

**UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR**



**TRABAJO FIN DE GRADO
GRADO EN INGENIERÍA INFORMÁTICA: SISTEMAS DE LA
INFORMACIÓN**

**ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE
UN SISTEMA GESTOR DE ALMACÉN
ADAPTADO A LA LOGÍSTICA FMS
(FOREIGN MILITARY SALES)**

**AUTOR: ADRIÁN CEJUDO RUIZ
TUTORA: BELÉN RUÍZ MEZCUA**

Junio de 2012

Dímelo y lo olvidaré.

Enséñame y lo recordaré.

Involúcrame y lo aprenderé". Benjamín Franklin

1. Índice

Contenido

1. Índice	3
2. Índice Tablas	6
3. Índice de Figuras.....	14
4. Introducción	17
4.1. Introducción a la logística militar	17
4.2. Estándar militar. Introducción a Foreign Military Sales (FMS).....	18
4.3. Motivación.	18
4.4. Objetivos	19
4.5. Estructura del documento.....	19
5. Estado del Arte / Estado de la cuestión	21
5.1. Introducción	21
5.2. Estudio.....	21
5.2.1 Easy WMS.....	21
5.2.2 Herramienta GoldenSoft.....	23
5.2.3 Infor WMS 2000	25
5.3 Cuadros Descriptivos.....	28
5.4 Conclusiones.....	31
6. Catálogo de Requisitos de la aplicación propuesta.....	32
6.1 Requisitos De Usuario	32
6.1.1 Capacidad.....	32
6.1.2 Restricción	39
6.2 Requisitos Técnicos	42
7. Análisis del sistema.....	43
7.1 Elaboración del modelo de procesos.....	43
7.1.1 Obtención del Modelo de Procesos del Sistema	43
7.1.2 Especificación de Interfaces con otros Sistemas.....	59
7.2 Análisis de los casos de uso	61
7.2.2 Funcionalidades del sistema.	61
7.2.3 Especificación de Casos de Uso.....	63

7.2.4	Identificación de Clases Asociadas a un Caso de Uso	68
7.2.5	Descripción de la Interacción de Objetos.....	69
7.3	Elaboración del modelo de datos.....	74
7.3.1	Elaboración del Modelo Lógico de Datos.....	74
7.3.2	Especificación de Necesidades de Migración de Datos y Carga Inicial	76
7.4	Estructura digital de los documentos electrónicos.....	76
8	Diseño del sistema	84
8.1	Definición de la arquitectura del sistema	84
8.1.1	Definición de Niveles de Arquitectura	84
8.1.2	Especificación del Entorno Tecnológico.....	85
8.2	Requisitos Del Sistema	87
8.2.1	Funcionales.....	87
8.2.2	No Funcionales	104
8.3	Arquitectura del sistema	104
8.3.1	Modelo de clases.....	105
8.3.2	Identificación de clases	106
8.3.3	Diseño de Asociaciones y agregaciones.....	109
8.3.4	Identificación de atributos de las clases	111
8.3.5	Identificación y Descripción de las Operaciones	117
8.4	Diseño de la arquitectura de módulos del sistema.....	126
8.4.1	Diseño de Módulos del Sistema y comunicación entre ellos.....	126
8.5	Diseño físico de datos	127
8.5.1	Diseño del Modelo Físico de Datos	127
8.5.2	Especificación del Script De Creación Almacén EE.UU.....	129
8.6	Diseño De La Interfaz	130
8.7	Alternativas de Diseño	135
8.7.1	Gestor De Base De Datos	135
8.7.2	Sistema Operativo	135
8.7.3	Aplicación	136
8.7.4	Interfaces De La Aplicación	136
9	Documento De Pruebas	137
9.1	Especificación De Pruebas Unitarias	137

9.1.1	Pruebas De Caja Blanca	137
9.1.2	Pruebas De Caja Negra	145
9.2	Especificación De Pruebas De Integración.....	158
9.3	Especificación De Pruebas De Inserción De Datos.....	158
10	Implantación y explotación	159
10.1	Instalación	159
10.1.1	Cliente	159
10.1.2	Servidor	160
11	Gestión del Proyecto	161
11.1	Introducción	161
11.2	Metodología y ciclo de vida	161
11.3	Estimación de recursos temporales	164
11.4	Estimación de recursos económicos	164
11.4.1	Recursos materiales.....	164
11.4.2	Recursos humanos	164
11.4.3	Costes totales	165
11.5	Gestión Del Riesgo	167
11.5.1	Riesgos Posibles.....	168
11.6	Plan del proyecto	172
11.6.1	Fases del proyecto.....	172
11.6.2	Diagrama De Gantt	176
11.7	Herramientas.....	179
11.7.1	Microsoft Windows 7	179
11.7.2	Integración con Internet Information Services	180
11.7.3	Microsoft Visual Studio 2010	181
11.7.4	Microsoft Office 2010.....	186
11.7.5	Microsoft Visio 2010	186
11.7.6	Aris Express	186
11.7.7	SQL Server 2008	187
11.7.8	Comparativa Sql Server y Mysql.	189
12	Conclusiones y trabajos futuros.....	190
12.1	Conclusiones.....	190

12.2	Trabajos futuros	191
13	Resumen en Inglés.....	192
14	Referencias	200
15	Anexo A Trazabilidad Requisitos Usuario/Software	201
16	Anexo B Glosario.....	204
17	Anexo C Script Creación Base De Datos EE.UU.....	207
18	Anexo D Script Creación Base De Datos Cliente	223

2. Índice Tablas

1	Cuadro Descriptivo Easy WMS	28
2	Cuadro Descriptivo GoldenSoft.....	29
3	Cuadro Descriptivo Infor	30
4	Requisito Capacidad UC-0001	32
5	Requisito Capacidad UC-0002	33
6	Requisito Capacidad UC-0003	33
7	Requisito Capacidad UC-0004	33
8	Requisito Capacidad UC-0005	34
9	Requisito Capacidad UC-0006	34
10	Requisito Capacidad UC-0007	34
11	Requisito Capacidad UC-0008	35
12	Requisito Capacidad UC-0009	35
13	Requisito Capacidad UC-0010	35
14	Requisito Capacidad UC-0011	36
15	Requisito Capacidad UC-0012	36
16	Requisito Capacidad UC-0013	36
17	Requisito Capacidad UC-0014	37
18	Requisito Capacidad UC-0015	37
19	Requisito Capacidad UC-0016	37
20	Requisito Capacidad UC-0017	38
21	Requisito Capacidad UC-0019	38
22	Requisito Capacidad UC-0019	38

23 Requisito Capacidad UC-0020	39
24 Requisito Restricción UR-0021	39
25 Requisito Restricción UR-0022	39
26 Requisito Restricción UR-0023	40
27 Requisito Restricción UR-0024	40
28 Requisito Restricción UR-0025	40
29 Requisito Restricción UR-0026	41
30 Requisito Restricción UR-0027	41
31 Requisito Restricción UR-0028	41
32 Requisito Técnico Rq_Tec_1.....	42
33 Requisito Técnico Rq_Tec_2.....	42
34 Requisito Técnico Rq_Tec_3.....	42
35 Requisito Técnico Rq_Tec_4.....	43
36 Requisito Técnico Rq_Tec_5.....	43
37 Caso De Uso Consultas FMS	63
38 Caso De Uso Gestión De Incidencias	63
39 Caso De Uso Recepciones	64
40 Caso De Uso Pedidos Enviados.....	64
41 Caso De Uso Pedidos Pendientes	64
42 Caso De Uso Pedidos No Procesados	64
43 Caso De Uso Gestión De Caja	65
44 Caso De Uso Gestión De Contenedor.....	65
45 Caso De Uso Recepción De La Mercancía	65
46 Caso De Uso Consolidación	65
47 Caso De Uso Envío	66
48 Caso De Uso Alta Usuario	66
49 Caso De Uso Habilitar Usuario	67
50 Caso De Uso Deshabilitar Usuario.....	67
51 Caso De Uso Eliminar Usuario	67
52 Requisito Sistema FSR -01	87
53 Requisito Sistema FSR -02	88
54 Requisito Sistema FSR -03	88

55 Requisito Sistema FSR -04	88
56 Requisito Sistema FSR -05	89
57 Requisito Sistema FSR -06	89
58 Requisito Sistema FSR -07	90
59 Requisito Sistema FSR -08	90
60 Requisito Sistema FSR -09	90
61 Requisito Sistema FSR -10	91
62 Requisito Sistema FSR -11	91
63 Requisito Sistema FSR -12	91
64 Requisito Sistema FSR -13	92
65 Requisito Sistema FSR -14	92
66 Requisito Sistema FSR -15	92
67 Requisito Sistema FSR -16	93
68 Requisito Sistema FSR -17	93
69 Requisito Sistema FSR -18	94
70 Requisito Sistema FSR -19	94
71 Requisito Sistema FSR -20	94
72 Requisito Sistema FSR -21	95
73 Requisito Sistema FSR -22	95
74 Requisito Sistema FSR -23	95
75 Requisito Sistema FSR -24	96
76 Requisito Sistema FSR -25	96
77 Requisito Sistema FSR -26	97
78 Requisito Sistema FSR -27	97
79 Requisito Sistema FSR -28	97
80 Requisito Sistema FSR -29	98
81 Requisito Sistema FSR -30	98
82 Requisito Sistema FSR -31	99
83 Requisito Sistema FSR -32	99
84 Requisito Sistema FSR -33	99
85 Requisito Sistema FSR -34	100
86 Requisito Sistema FSR -35	100

87 Requisito Sistema FSR -36	101
88 Requisito Sistema FSR -37	101
89 Requisito Sistema FSR -38	101
90 Requisito Sistema FSR -39	102
91 Requisito Sistema FSR -40	102
92 Requisito Sistema FSR -41	103
93 Requisito Sistema FSR -42	103
94 Requisito Sistema NFSR -43.....	104
95 Requisito Sistema NFSR -44.....	104
96 Identificación Clase Pedido	106
97 Identificación Clase DS_Pedido	106
98 Identificación Clase Caja.....	106
99 Identificación Clase DS_Caja	107
100 Identificación Clase Cargamento.....	107
101 Identificación Clase DS_Cargamento	107
102 Identificación Clase Envío.....	108
103 Identificación Clase DS_Envío	108
104 Identificación Clase Usuario	108
105 Identificación Clase DS_Usuario.....	109
106 Asociaciones Calce Pedido	109
107 Asociaciones Clase Caja.....	109
108 Asociaciones Clase Cargamento.....	110
109 Asociaciones Clase Envío.....	110
110 Asociación Clase Usuario.....	110
111 Asociación Caja - Pedido	110
112 Asociación Cargamento Caja.....	111
113 Asociación Envío Cargamento	111
114 Atributos Clase Pedido	111
115 Atributos Clase DS_Pedido.....	112
116 Atributos Clase Caja	113
117 Atributos Clase DS_Caja	113
118 Atributos Calce Cargamento	114

119 Atributos Calce DS_Cargamento	114
120 Atributos Clase Envío	115
121 Atributos Clase DS_Envio	115
122 Atributos Clase Usuario	116
123 Atributos Clase DS_Usuario	116
124 Método Clase Pedido ObtenerDatosPedido	117
125 Método Clase Pedido DividirPedido	117
126 Método Clase Pedido RecepcionarPedido	117
127 Método Clase Pedido ConsolidarPedido	117
128 Método Clase Pedido RecepcionarReparable	117
129 Método Clase Pedido ConsolidarReparable	118
130 Método Clase Pedido RecepcionarPedidoCliente	118
131 Método Clase Pedido EnvioClienteFin	118
132 ObtenerPedidosEnviados	118
133 ObtenerPedidosNoProcesados	118
134 ObtenerPedidosPendientes	119
135 Método Clase Pedido AltaIncidencia	119
136 Método Clase Pedido ObtenerPedidosNoInsertados	119
137 Método Clase Pedido ObtenerIncidencias	119
138 Método Clase Caja AbrirCaja	119
139 Método Clase Caja CerrarCaja	119
140 Método Clase Caja CrearCaja	120
141 Método Clase Caja ObtenerDatosCaja	120
142 Método Clase Caja InsertarPedido	120
143 Método Clase Caja ObtenerUltimaCaja	120
144 Método Clase Caja ObtenerCajas	120
145 Método Clase Caja ObtenerDetallesCaja	121
146 Método Clase Caja SacarPedido	121
147 Método Clase Caja ObtenerCajasNoInsertadas	121
148 Método Clase Cargamento AbrirCargamento	121
149 Método Clase Cargamento CerrarCargamento	121
150 Método Clase Cargamento CrearCargamento	122

151 Método Clase Cargamento ObtenerDatosCargamento	122
152 Método Clase Cargamento InsertarCaja	122
153 Método Clase Cargamento ObtenerDetallesCargamento	122
154 Método Clase Cargamento SacarCaja	122
155 Método Clase Cargamento ObtenerCargamentosNoInsertados	122
156 Método Clase Envío CrearEnvío	123
157 Método Clase Envío ObtenerEnvío	123
158 Método Clase Envío InsertarCargamento	123
159 Método Clase Envío CerrarEnvío.....	123
160 Método Clase Envío ObtenerEnvíoNoConsolidado.....	123
161 Método Clase Envío ObtenerContenidoEnvío	124
162 Método Clase Envío SacarCargamento	124
163 Método Clase Envío ConsolidarEnvío.....	124
164 Método Clase Envío ObtenerPedidosEnvío	124
165 Método Clase Usuario AltaUsuario	124
166 Método Clase Usuario DeshabilitarUsuario	124
167 Método Clase Usuario EliminarUsuario	125
168 Método Clase Usuario CambiarContraseñaUsuario	125
169 Método Clase Usuario LoginUsuario.....	125
170 Método Clase Usuario ComprobarCaducidadContraseña	125
171 Método Clase Usuario ObtenerDatosUsuario.....	125
172 Diccionario Tabla Caja	127
173 Diccionario Tabla Cargamento	128
174 Diccionario Tabla Envío	128
175 Diccionario Tabla Incidencias	128
176 Diccionario Tabla Línea	128
177 Diccionario Tabla Pedido	129
178 Diccionario Tabla Pedido_Caja.....	129
179 Diccionario Tabla Usuarios.....	129
180 Prueba Caja Blanca 01.....	137
181 Prueba Caja Blanca 02	138
182 Prueba Caja Blanca 03.....	138

183 Prueba Caja Blanca 04	138
184 Prueba Caja Blanca 05	138
185 Prueba Caja Blanca 06	139
186 Prueba Caja Blanca 07	139
187 Prueba Caja Blanca 08	139
188 Prueba Caja Blanca 09	139
189 Prueba Caja Blanca 10	140
190 Prueba Caja Blanca 11	140
191 Prueba Caja Blanca 12	140
192 Prueba Caja Blanca 13	140
193 Prueba Caja Blanca 14	141
194 Prueba Caja Blanca 15	141
195 Prueba Caja Blanca 16	141
196 Prueba Caja Blanca 17	142
197 Prueba Caja Blanca 18	142
198 Prueba Caja Blanca 19	142
199 Prueba Caja Blanca 20	143
200 Prueba Caja Blanca 21	143
201 Prueba Caja Blanca 22	143
202 Prueba Caja Blanca 23	144
203 Prueba Caja Blanca 24	144
204 Prueba Caja Blanca 25	144
205 Prueba Caja Blanca 26	145
206 Prueba Caja Blanca 27	145
207 Prueba Caja Negra 01	145
208 Prueba Caja Negra 02	146
209 Prueba Caja Negra 03	146
210 Prueba Caja Negra 04	147
211 Prueba Caja Negra 05	147
212 Prueba Caja Negra 06	148
213 Prueba Caja Negra 07	148
214 Prueba Caja Negra 08	149

215 Prueba Caja Negra 09	149
216 Prueba Caja Negra 10	150
217 Prueba Caja Negra 11	150
218 Prueba Caja Negra 12	151
219 Prueba Caja Negra 13	151
220 Prueba Caja Negra 15	152
221 Prueba Caja Negra 16	152
222 Prueba Caja Negra 17	153
223 Prueba Caja Negra 18	153
224 Prueba Caja Negra 19	154
225 Prueba Caja Negra 20	154
226 Prueba Caja Negra 21	155
227 Prueba Caja Negra 22	155
228 Prueba Caja Negra 23	156
229 Prueba Caja Negra 24	156
230 Prueba Caja Negra 25	157
231 Prueba Caja Negra 26	157
232 Riesgo Cliente No Disponible	168
233 Riesgo Tareas Faltantes, Poco Esfuerzo, Retraso De Entregas.....	169
234 Riesgo Pérdida De Personas, Desmotivación, Falta De Decisiones, Desastre Natural.....	170
235 Tareas Documentación Inicial	172
236 Tareas Análisis Del Sistema	173
237 Tareas Diseño Del Sistema	173
238 Tareas Desarrollo De Pruebas	174
239 Tareas Desarrollo	174
240 Tareas Implantación Y Explotación	175
241 Tareas Documentación Final	175
242 Requisitos Mínimos Windows 7	180
243 Requisitos Software Mínimos Visual Studio 2010.....	182
244 Requisitos Hardware Mínimos Visual Studio 2010	183
245 Comparativa Eclipse - Microsoft Visual Studio.....	185
246 Requisitos Software Sql Server 2008	188

247 Requisitos Hardware Sql Server 2008	188
248 Comparativa Sql Server Y Mysql.....	189
251 Referencias.....	200
252 Glosario	204

3. Índice de Figuras

1 Funcionalidades Easy WMS.....	21
2 Panel De Control Easy WMS.....	21
3 Stock GoldenSoft.....	23
4 Panel De Control GoldenSoft	23
5 Control De Stock Infor WMS 2000	25
6 Funcionalidades Infor WMS 2000	25
7 Visualización 2 De Estanterías Infor WMS 2001	26
8 Visualización 1 De Estanterías Infor WMS 2001	26
9 Pedido Nueva Adquisición.....	45
10 Lógica De Procesos De Almacén.....	49
11 Flujo Material Nueva Adquisición Extranjero.....	51
12 Flujo Material Nueva Adquisición Cliente	52
13 Gestión Documental Reparable	54
14 Gestión Mercancía A Reparar	55
15 Gestión Almacenes Reparable	57
16 Flujo Material Reparable	58
17 Interfaz De Comunicación Entre Sistemas	59
18 Especificación De La Comunicación Del Almacén	60
19 Diagrama De Casos De Uso Operario	63
20 Diagrama De Casos De Uso Administrador	66
21 Diagrama De Secuencia Consulta De Pedidos.....	69
22 Diagrama De Secuencia Gestión De Cajas.....	69

23 Diagrama De Secuencia Consulta De Cargamentos	70
24 Diagrama De Secuencia Consolidación De Pedidos	70
25 Diagrama De Secuencia Envíos	71
26 Diagrama De Secuencia Recepción De Pedidos	71
27 Diagrama De Secuencia De Alta Incidencias	71
28 Diagrama De Secuencia De Alta Usuarios	72
29 Diagrama De Secuencia Deshabilitar Usuarios	72
30 Diagrama De Secuencia Habilitar Usuarios	73
31 Diagrama De Secuencia Eliminar Usuarios.....	73
32 Modelo Entidad-Relación De La Base De Datos.....	74
33 Modelo Relacional De La Base De Datos.....	75
34 Identificación De Las Líneas FMS	76
35 Niveles De La Arquitectura Del Sistema	84
36 Entorno Tecnológico Desarrollo En 3 Capas	85
37 Arquitectura Software Del Sistema.....	86
38 Diagrama De Clases.....	105
39 Desarrollo De La Arquitectura En 3 Capas	126
40 Comunicación Bases De Datos Aplicación.....	130
41 Interfaz De Login	131
42 Interfaz Alta Usuario	131
43 Interfaz Cambio De Password	132
44 Interfaz Búsqueda De Pedidos	132
45 Interfaz Abrir Caja	133
46 Interfaz Insertar Pedido.....	133
47 Interfaz Cerrar Caja	134
48 Interfaz Detalle Cargamento	134
49 Interfaz Menu Superior	134
50 Calidad Del Software	161
51 Ciclo De Vida.....	163
52 Costes De Las Tareas	166
53 Diagrama De Gantt General	176
54 Diagrama De Gantt Documetación Inicial	176

55 Diagrama De Gantt Análisis Del Sistema	177
56 Diagrama De Gantt Diseño Del Sistema	177
57 Diagrama De Gantt Desarrollo Y Pruebas	178
58 Gantt Documentación Final	179
59 Proceso FMS Resumen	194
60 Diagrama E/R Resumen.....	197
61 Interfaz Ejemplo Resumen	199
62 Gantt Resumen.....	199

4. Introducción

4.1. Introducción a la logística militar

La logística militar es parte de la ciencia y arte de la guerra, y como ella, ha sido parte de la historia de la humanidad, con la cual ha evolucionado, y se ha refinado hasta convertirse en una ciencia de aplicación a diferentes procesos de apoyo a las Fuerzas Operativas.

LA LOGÍSTICA MILITAR se define como «la parte del arte de la guerra que tiene por objeto proporcionar a las Fuerzas Armadas los medios necesarios para satisfacer adecuadamente las exigencias de la guerra».

La definición establece una función: PROPORCIONAR LOS MEDIOS. Los medios requeridos por las Fuerzas Militares, que son: el personal, es decir, los medios humanos; el material, tales como medios físicos de combate y apoyo; y los servicios, todo tipo de actividades que generen beneficios directos o indirectos a la conducción de la guerra.

Asimismo, la definición impone una condición: que dichos medios SATISFAGAN ADECUADAMENTE las exigencias de la guerra. Esta condición es tan compleja como la guerra misma, y pueden identificarse condiciones como la **OPORTUNIDAD**, en razón de una necesidad específica, esta no puede ser atendida antes de conocerse y después de requerirse, pues podría ser funesto; el **LUGAR** donde se produce la necesidad y donde debe ser satisfecha; es el sitio geográfico donde se presenta la exigencia; la **CANTIDAD**, ¿qué tanto de algo se necesita? Esta cantidad debe ser la dosis justa a la exigencia; la **CALIDAD** es, dentro de la multitud de productos y servicios, aquel que reúne las especificaciones apropiadas para la exigencia requerida. Juntando todas las características citadas anteriormente se puede asegurar cualquier exigencia necesitada.

El ciclo logístico, la determinación de las necesidades, la obtención y la distribución también pueden ser aplicados a la empresa privada, tanto como los elementos funcionales logísticos se constituyen en empresas y los principios de la logística son interpretados a la luz de las organizaciones empresariales, porque las instituciones militares como la Armada Nacional pueden y deben ser administradas con conceptos gerenciales de uso universal.

Para acabar dicha introducción, recordamos una cita de Julio César, emperador, líder militar y político de la época romana. En este sentido afirma que una campaña militar debe centrarse en “llegar, ver y vencer” [Julio César, 100 a.C, Emperador Romano].

Así, la logística militar es la encargada de darle vida a la función de “llegar”; las fuerzas deben llegar al lugar en el momento adecuado, con las condiciones de abastecimiento necesarias para poder realizar las tareas que les fueron asignadas. Y en éste sentido es donde se mueve la logística militar.

En este proyecto se va a implantar las herramientas necesarias para automatizar la logística militar de envío de pedidos siguiendo los estándares definidos.

Palabras claves: Logística militar, cielo logístico, logística de producción y consumo.

La logística militar y sus aplicaciones en la logística empresarial Eduardo Posada Zamudio[en línea] www.dsca.mil [Consulta:12/12/2011]

4.2. Estándar militar. Introducción a Foreign Military Sales (FMS)

THE FOREIGN MILITARY SALES (FMS) es un programa o estándar propio de Estados Unidos para distribuir artículos de defensa o servicios a otras naciones u organizaciones internacionales.

Bajo FMS, el gobierno estadounidense procura mercancías de defensa y servicios al cliente extranjero.

Existe un acuerdo con numerosos países para el intercambio de mercancías con Estados Unidos bajo éste formato, entre ellos se incluye España.

La Agencia de Cooperación de Seguridad de Defensa (DSCA) administra el programa FMS para el Departamento de defensa (DoD).

Actualmente unos 160 países son partícipes en éste programa.

4.3. Motivación.

Como se ha recogido, la logística militar está perfectamente definida y estandarizada y los procesos que gestionen las mismas deben contemplar estos estándares y cumplirlos eficazmente. Por ello y para satisfacer la logística militar y el estándar FMS, se desarrolla en este proyecto una herramienta que permita gestionar, distribuir y realizar una trazabilidad y gestión de la misma centrándonos en el presente trabajo, en las fases de gestión y distribución de la mercancía dentro de los almacenes y en la distribución y transporte los materiales tanto por tierra, mar o aire.

El fin es obtener un Software capaz de gestionar toda la mercancía militar bajo el programa de FMS. De tal forma que se pueda generar una trazabilidad de los pedidos con el fin de obtener la información necesaria para saber dónde se sitúa cualquier mercancía en tiempo real.

Si un país realiza una compra a Estados Unidos bajo éste formato (por ejemplo un barco o armamento), es de vital importancia saber que todas las piezas o mercancías están bien recogidas en una base de datos para poder acceder a ellas fácilmente en el almacén de destino, a parte de la trazabilidad comentada anteriormente o una posible gestión de incidencias o informes.

4.4. Objetivos

A partir de las motivaciones el objetivo es poder dar soporte a la gestión de mercancías de forma clara y sencilla implementando un entorno estable de un sistema logístico centrado en la gestión de almacén que responda a las exigencias citadas anteriormente bajo el programa FMS.

Por tanto el objetivo principal es la realización de un Software capaz de gestionar un almacén siguiendo los estándares definidos del FMS.

Como objetivos secundarios, dicho proyecto servirá para poder realizar una trazabilidad de la vida de los envíos de los pedidos a través de las líneas FMS generadas en el almacén, coordinando toda la información vía internet.

4.5. Estructura del documento

El siguiente documento está dividido en varias secciones o grupos:

- Estado Del Arte/ Estado De La Cuestión: sección en la cual se van a evaluar las herramientas y empresas relacionadas con el sector logístico.
- Catálogo de Requisitos donde se recogerán los requisitos de la aplicación tanto de usuario como de capacidad y técnicos.
- Análisis del sistema donde se desarrollará un estudio del estándar FMS Americano junto con un exhaustivo modelo de procesos que ayude a entenderla elaborando el diagrama de casos de usos junto con el modelo de datos.

- Diseño del sistema en la cual se definirá la arquitectura del sistema junto con el diseño del diagrama de clases identificando todos sus atributos, relaciones y métodos. Éste apartado acabará de tal forma que se definirá el diseño físico de datos junto con los Scripts necesarios para la creación de las bases de datos, finalizando con un diseño de las interfaces y una reseña a las alternativas de diseño que se podrían haber realizado.
- Documento De Pruebas en el cual se definirán las pruebas unitarias (Caja Blanca y Caja Negra), pruebas de integración y pruebas de inserción de datos para comprobar que los scripts antes comentados funcionan correctamente.
- Realización de prototipo y diseño de la interfaz.
- Implantación y explotación: se explicará los programas y servidores necesarios para instalar dicha aplicación y ponerla en funcionamiento así como las versiones que deben de ser instaladas y configuradas.
- Gestión del proyecto en el cual se reflejará la gestión del tiempo y recursos empleados visualizándose el diagrama de Gantt obtenido. En ésta sección se han incluido también las herramientas utilizadas para el desarrollo e implantación de la aplicación obteniendo cuadros comparativos con las posibles alternativas.
- Conclusiones y trabajos futuros: se expondrán las conclusiones obtenidas en el desarrollo de todo el proyecto y las posibles actualizaciones, mejoras tecnológicas y nuevas funcionalidades que se podrían implementar.
- Resumen en Inglés del proyecto.
- Referencias.
- Anexo: Glosario.

5. Estado del Arte / Estado de la cuestión

5.1. Introducción

En este apartado se van a evaluar las herramientas y empresas relacionadas con el sector logístico.

En la actualidad existen diversas aplicaciones implementados en gran variedad de lenguajes que desempeñan una función de gestor de almacén tal como en éste proyecto se quiere reflejar.

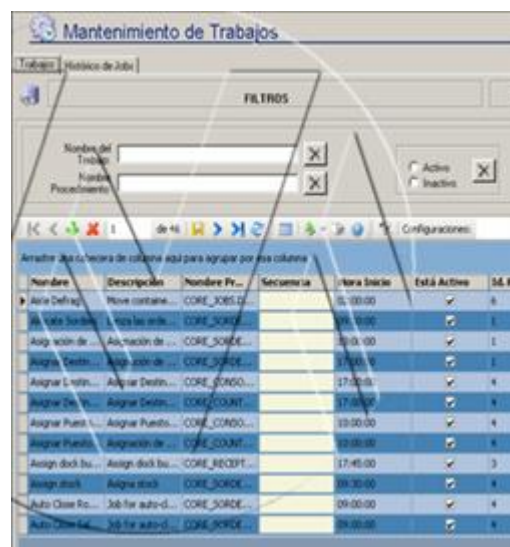
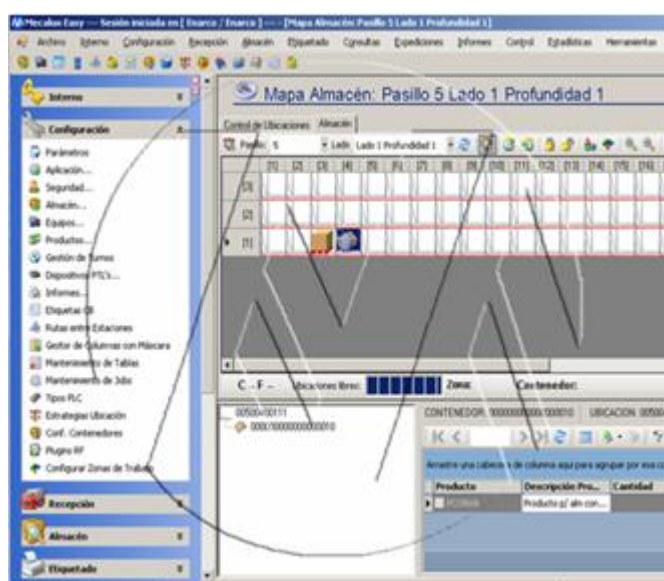
A continuación se muestra un estudio de las tecnologías y aplicaciones que son usadas en la actualidad para llevar a cabo un proceso de almacén.

Para ello se han detectado las herramientas que utilizan diferentes empresas que implementan parte de las funcionalidades propuestas en el presente proyecto.

5.2. Estudio

5.2.1. Easy WMS

Easy WMS: Software privado de la empresa Mecalux Esmena.



1 Funcionalidades Easy WMS

2 Panel De Control Easy WMS

Las funcionalidades más importantes abordadas por la herramienta son:

- Creación de contenedores de tipo Europaleta y Universal (Ambos Palets) y EuroCaja (Contenedor) con su código único con medidas y foto si se desea.

- Creación y configuración de etiquetas tanto de contenedores como mercancías. Es posible aumentar o disminuir el número de etiquetas por hoja.
- Gestión de equipamientos: carretillas o carros para el transporte de la mercancía o contenedores pudiendo realizar restricciones por tipo de mercancía o contenedor.
- Gestión de estanterías: configuración del tipo, número de columnas, medidas de la fila.
- Gestión del recorrido de una mercancía (dentro del almacén).
- Gestión de recepciones y entradas en el almacén. Confirmación del pedido enviado y recepcionado con la posibilidad de incluir notas o comentarios en los pedidos.
- Gestión de expediciones y órdenes de salida.
- Creación de cuentas de cliente con la posterior configuración y visualización de datos.
- Gestión y creación de informes de tracking y pedidos.
- Control de expediciones y órdenes de salida.
- Alta de artículos con sus características con la posibilidad de añadir descripciones u observaciones y creación de nuevas medidas.
- Uso de plantillas para la carga de mercancías mediante formato Excel.
- Control de proveedores y stock de sus productos.
- Gestión de cuentas de usuario y roles de administrador, cliente y proveedor.

Sin embargo las debilidades de la herramienta son:

- El código de barras no es configurable. Sólo es posible usar el código de barras que viene por defecto sin poder usar otra fuente o código. Actualmente existen muchos tipos de códigos como los QR, deberían recoger el hecho de poder seleccionar el tipo de código para que su lectura mediante PDA o Smartphone sea rápida y sencilla.
- Las etiquetas se imprimen mediante impresora láser. Es necesario configurar la impresora y el papel para poder realizar la impresión correctamente. Si se tuviera una configuración estándar se podría imprimir incluso mediante una impresora Zebra.
- El formato de los códigos o claves primarias tanto de cajas, como mercancías y elementos no está regido por ningún estándar.
- Tampoco es posible realizar una trazabilidad de los pedidos realizados de tal forma que se vaya actualizando la situación actual del pedido diariamente.
- No se recogen incidencias o cambios tanto de embalajes como elementos que han sido modificados una vez situados.
- Es una aplicación de escritorio.

Por lo demás es un Software muy recomendable para la gestión y configuración de un almacén o nave, ya que sus características te permiten controlar hasta las baldas de las estanterías numerándolas, de tal forma que se tenga un acceso rápido y fiable.

Mecalux, Esmena. "Solucione De Almacenaje" [en línea]. Diciembre 2011, www.mecalux.es/wms

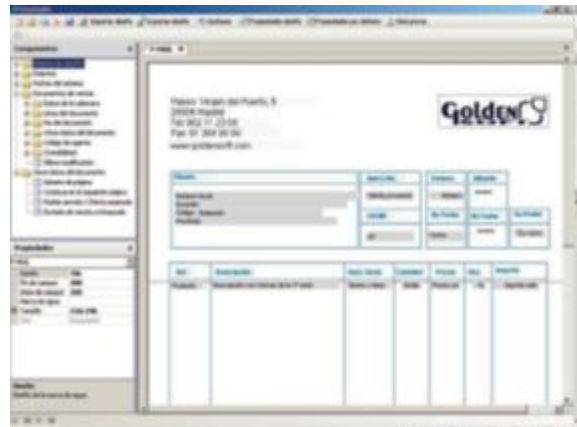
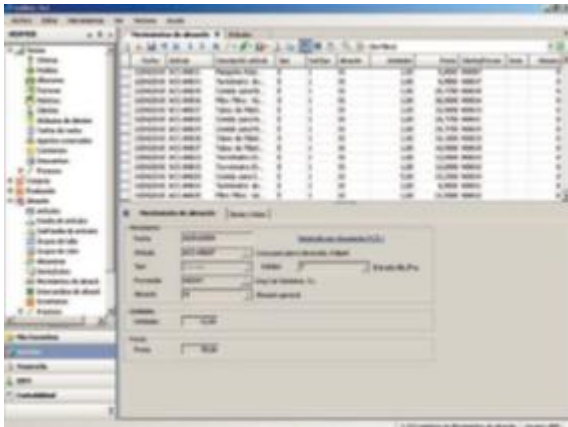
Esta herramienta no cumple los objetivos planteados por el presente proyecto en:

- No se aplica el estándar FMS (Foreign Military Sales).
- No hay una clara identificación de pedidos y divisiones.
- No es una aplicación web.

Otra herramienta destacable es:

5.2.2 Herramienta GoldenSoft

Sistema contable y gestor de almacén con múltiples funciones.



3 Stock GoldenSoft

4 Panel De Control GoldenSoft

Funcionalidades destacadas:

- Cálculo del stock de los productos en el almacén de forma automática incluyendo su seguimiento.
- Menú integrado en un único punto consiguiendo una gran facilidad de uso.
- Seguimiento de todo tipo de información. Se recogen desde pedidos, presupuestos, facturas de gestión, cuentas contables a e-mails o una relación de llamadas telefónicas.
- Generación automática de pedidos de compra

- Visualización de información en formato lista, ficha o ambos.
- Integración con Microsoft Excel y Outlook.
- Incluye soporte y ayuda on-line.
- Gestión y creación de filtros personalizados, haciendo más rápida la tarea de buscar cualquier elemento.
- Avisos intranet. Con dicha mensajería interna se pueden enviar avisos u observaciones a otros usuarios, grupos o departamentos.
- Elaboración de informes en formato Pdf.
- Generación de previsiones de forma automática, tanto puntual como periódica.
- Histórico de acciones realizadas.
- Control individual de los artículos o servicios según su fecha de fabricación y caducidad, permitiendo supervisar la trazabilidad del producto
- Generación de pedidos y proveedores.
- Automatización del pedido de compras necesario para servir la mercancía solicitada por el cliente.
- Factura Electrónica con formato Factura-E.
- Posibilidad de enlace con Sistema EDI.

Sin embargo las debilidades de la herramienta son:

- Es una aplicación de escritorio.
- No sabemos con exactitud el modo de trabajo que se tiene en el almacén. Si se imprimen etiquetas para relacionar elementos y si su lectura es mediante PDA u otros dispositivos.
- Tampoco se puede llegar a saber cómo se identifican los elementos o si llevan algún tipo de estándar en los códigos de cualquier elemento.

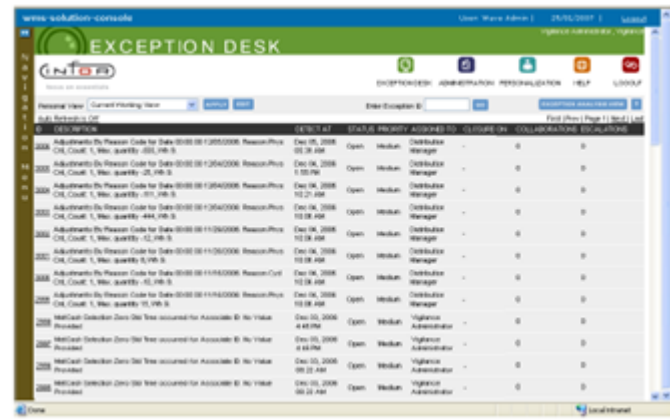
GoldenSoft. "Gestión Comercial Y Contable" [en línea]. Diciembre 2011, <http://www.goldensoft.com>

Esta herramienta no cumple los objetivos planteados por el presente proyecto en:

- No se aplica el estándar FMS (Foreign Military Sales).
- No hay una clara identificación de pedidos y divisiones.
- No es una aplicación web.
- No existe una gestión clara de cajas o cargamentos en los cuáles insertar los pedidos.

5.2.3 Infor WMS 2000

Infor es otra de las herramientas destacables y es el proveedor de software empresarial de más rápido crecimiento en la industria además de ser el tercer suministrador de software de gestión empresarial. Además merece un punto mayor de estudio al ser una empresa importante que colabora con el Ministerio de Defensa de España.



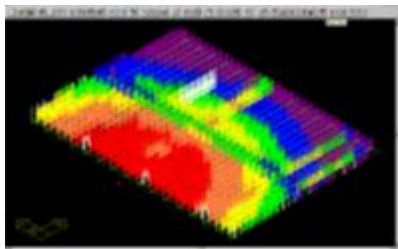
5 Control De Stock Infor WMS 2000

6 Funcionalidades Infor WMS 2000

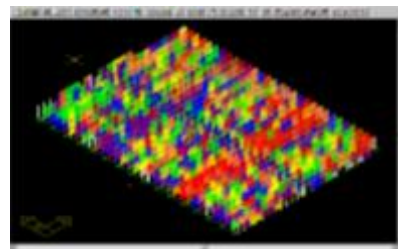
Funcionalidades destacadas:

- Entrada de mercancías recogiendo :
 - o Fecha de entrada.
 - o Diversos modos de recepción
 - o Impresión de códigos de barras por código EAN 128, Postnet: US, Code 39 Health(HIBCC), PDF817(Fedex), UPC Retail, MaxiCode(UPS).
 - o Intercambio de información de las mercancías con mensajería EDI.
- Configuración avanzada para optimizar la utilización del espacio de almacenamiento y/o la productividad.
- Parámetros de producto: tamaño, peso, fechas de caducidad, control de lote combinados con parámetros de las ubicaciones: tamaño, capacidad volumétrica, peso permitido, apilabilidad y zonas sirven para elaborar una lista de ubicaciones potenciales.
- Las ubicaciones se analizan desde el punto de vista de la productividad y el aprovechamiento del volumen utilizable para escoger la ubicación más óptima.
- Gestión flexible de pedidos incluyendo su planificación, generación, mantenimiento y asignación.

- Parametrización del Picking (Situación de la mercancía y embalajes en el almacén):
 - Definición de contenedores y volúmenes por sección, agrupación de ubicaciones, tipo de pedido, cliente -prioridad-.
 - Listado, Etiquetas, y encontrado de mercancías por voz.
 - Control de secciones, volumen máximo peso máximo y tiempo máximo.
 - Captura de peso variable.
- Carga de la mercancía:
 - Planificación de citas de expedición de mercancía.
 - Consolidación y reorganización de los palés en la expedición: consolidación total, parcial
 - Tareas de carga de palés por lectura del código mediante PDA.
 - Verificación y cierre de la carga por PDA.
- Control de la reposición de las mercancías:
 - Puntos de reposición configurables.
 - Almacenamiento de palé o cajas restantes después de reposición.
 - Rellenado de ubicaciones.
- Gestión de inventarios:
 - Gestión de contenedores avanzada
 - Captura peso variable
 - Transferencias de inventario, movimientos y ajustes
 - Control y consulta del inventario en tiempo real.
 - Monitorización de fechas de caducidad y vida útil.
- Monitorización de actividades:
 - Identificación de actividades pendientes por tipo, prioridad, sección, etc. en tiempo real.
 - Revisión de las asignaciones una vez terminadas.
 - Despacho de tareas basado en prioridad.
- Gestión de recursos con control de asistencia, seguimiento de asignaciones, estimación de tiempos.
- “Slotting”: control avanzado de artículos identificados por criterios como; artículos de alta rotación o productos pesados ergonómicamente identificados.



8 Visualización 1 De Estanterías Infor WMS 2001



7 Visualización 2 De Estanterías Infor WMS 2001

- Visibilidad del Inventario en Tiempo Real para los tráiler y su contenido.
- Gestión de informes:
 - Repositorio listados estándares.
 - Listados gráficos.
 - Recepciones.
 - Control de Inventario.
 - SKUs.
 - Expediciones.
 - Ubicaciones.
 - Proveedores.

Sin embargo las debilidades de la herramienta son:

El producto contiene una gran funcionalidad de todo tipo para el control global del almacén.

Respecto a los otros Software estudiados anteriormente es posible que no tenga tanto control de ubicaciones tan restrictivas, pero su funcionalidad de “slotting” hace que no sea algo grave.

Al trabajar para el Ministerio de Defensa todas las funcionalidades propias de un gestor de almacén son recogidas, sin embargo (como en los anteriores estudios de las aplicaciones citadas anteriormente) no se contiene ningún tipo de estándar en el desarrollo de la gestión de identificadores.

Infor Enterprise Software Solutions [en línea]. Diciembre 2011, www.infor.com


Esta herramienta no cumple los objetivos planteados por el presente proyecto en:

- No se aplica el estándar FMS (Foreign Military Sales).

5.3 Cuadros Descriptivos

Easy WMS

1 Cuadro Descriptivo Easy WMS

				
Producto	Promoción	Variedad	Posicionamiento	Servicio
Aplicación de escritorio para la gestión de almacenes.	Vía anuncios Webs y publicidad a empresas del sector.	Centrada en la zona de almacén o nave.	Al no poder acceder a las empresas que usan sus servicios, no se puede establecer un posicionamiento en el mercado actual.	Amplia gama de gestión de proveedores, cajas, elementos, mercancías y stock.
<p>Fortalezas: Tal como se ha comentado anteriormente, dispone de un amplio despliegue de posibilidades para la situación de mercancías a niveles muy restrictivos. Se dispone de una numeración única de cajas y mercancías mediante código de barras de tal forma que el acceso a su información es rápido y preciso.</p>				
<p>Debilidades: No es posible seleccionar un código de barras distinto al que viene por defecto, lo que puede hacer que ciertos dispositivos no puedan leerlo. Tampoco es posible configurar la impresora en la que se quieren imprimir dichos códigos de barras. Además de no tener una gestión de incidencias u observaciones en el propio almacén.</p> <p>También se puede incluir como punto débil que la aplicación es de escritorio y no web, de tal forma que debe ser instalada en todas las máquinas que vayan hacer uso de él.</p> <p>La debilidad más importante es que no siguen un estándar para recoger los identificadores de embalajes o elementos</p>				

GoldenSoft


Requiere aprendizaje para poder sacar todo el potencial de la herramienta.



Producto	Promoción	Variedad	Posicionamiento	Servicio
Aplicación de escritorio que engloba gestión de almacén, económica y facturas	Vía anuncios Webs y publicidad a empresas del sector.	Engloba el almacén, trazabilidad de las mercancías y total gestión económica con proveedores y facturas online.	Al no poder acceder a las empresas que usan sus servicios, no se puede establecer un posicionamiento en el mercado actual.	Amplia gama de gestión de proveedores, cajas, elementos, mercancías y stock junto con sus facturas online.
<p>Fortalezas: Engloba mucha variedad de funciones respecto a la gestión del almacén. A parte de gestionar los embalajes y mercancías junto con su situación, también se incluye la gestión de creación de facturas online de dichos elementos junto con toda la gestión de proveedores y pagos. Además de poder llevar un control de producción de elementos para control de stock.</p>				
<p>Debilidades: No sabemos con exactitud que tipo de códigos se usan tanto en mercancías como embalajes o facturas, si llevan algún estándar o no.</p> <p>El modo de trabajo tampoco es apreciable saberlo sólo con dicho sistema, no es posible averiguar si el trato de las mercancías es por orden de llegada u otro sistema de colas.</p> <p>Aunque la interfaz del sistema es intuitiva y fácil de manejar, es necesario hacer un breve estudio de la aplicación para conocer todas las funcionalidades incluyendo toda la gestión económica.</p>				

2 Cuadro Descriptivo GoldenSoft

3 Cuadro Descriptivo Infor

				
Producto	Promoción	Variedad	Posicionamiento	Servicio
Aplicación web de gestión de almacén.	Software utilizado en el Ministerio de Defensa Español.	Engloba el almacén y la trazabilidad junto con un gran sistema de situación de mercancías y embalajes.	Gran posicionamiento al trabajar para un Ministerio.	Buena gestión de almacén controlando todo tipo de información útil y clasificable.
<p>Fortalezas: La gran fortaleza de dicho Software se encuentra en su característica denominada slotting, capaz de obtener de forma gráfica los elementos situados en las estanterías buscando por su atributo.</p> <p>De ésta forma optimiza el coste reduciendo el tiempo de realización aumentando la utilización de activos y la productividad de los operadores.</p>				
<p>Debilidades:</p> <p>Aunque se detalla que la aplicación tiene una generación y control de informes, no sabemos con exactitud cómo es dicha gestión y su expedición.</p> <p>No uso de estándar en la totalidad de la gestión del almacén incluyendo identificación de elementos, embalajes o pedidos.</p>				

5.4 Conclusiones

Las tres herramientas estudiadas al ser similares en la gestión propia de un almacén, tenemos 3 ejemplos de cómo cada Software es distinto según el cliente y el uso que quiera dar de él.

La primera (Easy WMS) está más centrada a un control detallado del almacén, gestionando la situación de la mercancía de forma muy restrictiva con el uso de estanterías y baldas. Éste Software está destinado al uso de pequeños establecimientos para tener un control de sus productos o elementos controlando su stock en todo momento pudiendo dar de alta nuevos elementos con gran variedad de atributos.

Sin embargo el segundo estudio (GoldenSoft) está más dedicado al control de empresas con mayor número de almacenes y no tanto el control de la situación de las mercancías si no la gestión económica y mayor control de proveedores y facturas. En ésta aplicación a parte del control del inventario se puede realizar un mayor control de todas las facturas englobando una mayor y detallada gestión económica.

El último Software está más definido para el ahorro y costes de la empresa. Al estar desarrollado para el uso público, es de primera importancia el ahorro en costes y uso. Por eso dicha aplicación se centra más en la facilidad de uso y trabajo para los operarios que lo utilicen.

6. Catálogo de Requisitos de la aplicación propuesta

En este apartado se detallaran los requisitos de usuario que será necesario implementar en la aplicación para satisfacer las necesidades del usuario. Para la realización de estos requisitos se ha contado con la colaboración del cliente que en este caso ha sido Raúl Díaz Gutiérrez de Sli Uti.

6.1. Requisitos De Usuario

A continuación se muestran los requisitos de usuario ó capacidad obtenidos.

El formato de los requisitos será el siguiente:

ID: Identificador	Tipo: Tipo Requisito	Necesidad: Baja/Media/Alta
Título: Título del requisito		
Fuente: Fuente del requisito		
Descripción: Descripción corta del requisito		
Estabilidad: Baja/Media/Alta		
Verificabilidad: Baja/Media/Alta		

6.1.1 Capacidad

4 Requisito Capacidad UC-0001

ID: UC – 0001	Tipo: Capacidad	Necesidad: Alta
Título: Identificación De Roles		
Fuente: Cliente		
Descripción: Se tendrán 2 roles en la aplicación con sus distintos privilegios. El Rol operativo y administrador podrán acceder a toda la funcionalidad de la aplicación.		
Estabilidad: Alta		
Verificabilidad: Alta		

5 Requisito Capacidad UC-0002

ID: UC – 0002	Tipo: Capacidad	Necesidad: Alta
Título: Alta De Cargamentos		
Fuente: Cliente		
Descripción: Se podrá dar de alta un cargamento por parte del operario.		
Estabilidad: Alta		
Verificabilidad: Alta		

6 Requisito Capacidad UC-0003

ID: UC – 0003	Tipo: Capacidad	Necesidad: Alta
Título: Cerrar Cargamentos		
Fuente: Cliente		
Descripción: Se podrá cerrar un cargamento por parte del operario.		
Estabilidad: Alta		
Verificabilidad: Alta		

7 Requisito Capacidad UC-0004

ID: UC – 0004	Tipo: Capacidad	Necesidad: Alta
Título: Abrir Cargamentos		
Fuente: Cliente		
Descripción: Se podrá abrir un cargamento por parte del operario.		
Estabilidad: Alta		
Verificabilidad: Alta		

8 Requisito Capacidad UC-0005

ID: UC – 0005	Tipo: Capacidad	Necesidad: Alta
Título: Alta De Caja		
Fuente: Cliente		
Descripción: El operario podrá crear una nueva caja		
Estabilidad: Alta		
Verificabilidad: Alta		

9 Requisito Capacidad UC-0006

ID: UC – 0006	Tipo: Capacidad	Necesidad: Alta
Título: Abrir Caja		
Fuente: Cliente		
Descripción: El operario podrá abrir una caja cerrar		
Estabilidad: Alta		
Verificabilidad: Alta		

10 Requisito Capacidad UC-0007

ID: UC – 0007	Tipo: Capacidad	Necesidad: Alta
Título: Cerrar Caja		
Fuente: Cliente		
Descripción: El operario podrá cerrar una caja abierta para su consolidación posterior.		
Estabilidad: Alta		
Verificabilidad: Alta		

11 Requisito Capacidad UC-0008

ID: UC – 0008	Tipo: Capacidad	Necesidad: Alta
Título: Insertar Pedido en Caja		
Fuente: Cliente		
Descripción: El operario podrá insertar un pedido en una caja abierta		
Estabilidad: Alta		
Verificabilidad: Alta		

12 Requisito Capacidad UC-0009

ID: UC – 0009	Tipo: Capacidad	Necesidad: Alta Muy Alta
Título: Desasignar Pedido en Caja		
Fuente: Cliente		
Descripción: El operario podrá sacar un pedido en una caja abierta		
Estabilidad: Alta		
Verificabilidad: Alta		

13 Requisito Capacidad UC-0010

ID: UC – 0010	Tipo: Capacidad	Necesidad: Muy Alta
Título: Insertar Caja en Cargamento		
Fuente: Cliente		
Descripción: El operario podrá insertar cajas cerradas en un cargamento abierto.		
Estabilidad: Alta		
Verificabilidad: Alta		

14 Requisito Capacidad UC-0011

ID: UC – 0011	Tipo: Capacidad	Necesidad: Muy Alta
Título: Desasignar Caja de Cargamento		
Fuente: Cliente		
Descripción: El operario podrá desasignar cajas cerradas de un cargamento.		
Estabilidad: Alta		
Verificabilidad: Alta		

15 Requisito Capacidad UC-0012

ID: UC – 0012	Tipo: Capacidad	Necesidad: Muy Alta
Título: Asignar Cargamento en un Envío.		
Fuente: Cliente		
Descripción: El operario podrá asignar cargamentos cerrados a un envío.		
Estabilidad: Alta		
Verificabilidad: Alta		

16 Requisito Capacidad UC-0013

ID: UC – 0013	Tipo: Capacidad	Necesidad: Muy Alta
Título: Desasignar Cargamento de un Envío.		
Fuente: Cliente		
Descripción: El operario podrá desasignar cargamentos cerrados de un envío que no haya sido enviado.		
Estabilidad: Alta		
Verificabilidad: Alta		

17 Requisito Capacidad UC-0014

ID: UC – 0014	Tipo: Capacidad	Necesidad: Muy Alta
Título: Recepciones		
Fuente: Cliente		
Descripción: El operario podrá insertar pedidos que lleguen en el almacén.		
Estabilidad: Alta		
Verificabilidad: Alta		

18 Requisito Capacidad UC-0015

ID: UC – 0015	Tipo: Capacidad	Necesidad: Muy Alta
Título: Cerrar Envíos		
Fuente: Cliente		
Descripción: El operario podrá cerrar un envío para su traslado posterior.		
Estabilidad: Alta		
Verificabilidad: Alta		

19 Requisito Capacidad UC-0016

ID: UC – 0016	Tipo: Capacidad	Necesidad: Alta
Título: Consulta De Pedidos No Procesados		
Fuente: Cliente		
Descripción: El operario podrá consultar los pedidos que todavía no han consolidados.		
Estabilidad: Alta		
Verificabilidad: Alta		

20 Requisito Capacidad UC-0017

ID: UC – 0017	Tipo: Capacidad	Necesidad: Alta
Título: Consulta De Pedidos No Recepcionados		
Fuente: Cliente		
Descripción: El cliente podrá consultar pedidos que todavía no han llegado al almacén.		
Estabilidad: Alta		
Verificabilidad: Alta		

21 Requisito Capacidad UC-0019

ID: UC – 0018	Tipo: Capacidad	Necesidad: Alta
Título: Crear Envío		
Fuente: Cliente		
Descripción: El cliente podrá crear un envío para insertar cargamentos en él.		
Estabilidad: Alta		
Verificabilidad: Alta		

22 Requisito Capacidad UC-0019

ID: UC - 0019	Tipo: Capacidad	Necesidad: Media
Título: Gestión Usuarios		
Fuente: Cliente		
Descripción: El Administrador podrá añadir, eliminar o deshabilitar usuarios en el sistema.		
Estabilidad: Alta		
Verificabilidad: Alta		

23 Requisito Capacidad UC-0020

ID: UC - 0020	Tipo: Capacidad	Necesidad: Media
Título: Alta Incidencias		
Fuente: Cliente		
Descripción: La aplicación podrá añadir incidencias asociados al pedido.		
Estabilidad: Alta		
Verificabilidad: Alta		

6.1.2 Restricción

24 Requisito Restricción UR-0021

ID: UR – 0021	Tipo: Restricción	Necesidad: Muy Alta
Título: Estándar FMS		
Fuente: Cliente		
Descripción: La aplicación deberá registrarse por el estándar de uso FMS del ejército americano, se incluirán también el proceso de reparables.		
Estabilidad: Alta		
Verificabilidad: Alta		

25 Requisito Restricción UR-0022

ID: UR – 0022	Tipo: Restricción	Necesidad: Media
Título: Tecnología		
Fuente: Cliente		
Descripción: La aplicación soportará los navegadores más comunes y utilizados: Internet Explorer, Mozilla Firefox.		
Estabilidad: Alta		
Verificabilidad: Alta		

26 Requisito Restricción UR-0023

ID: UR – 0023	Tipo: Restricción	Necesidad: Alta
Título: Seguridad Contraseñas		
Fuente: Cliente		
Descripción: Las contraseñas caducarán en un periodo fijado.		
Estabilidad: Alta		
Verificabilidad: Alta		

27 Requisito Restricción UR-0024

ID: UR – 0024	Tipo: Restricción	Necesidad: Alta
Título: Seguridad de cifrado		
Fuente: Cliente		
Descripción: Las contraseñas se guardarán en formato md5		
Estabilidad: Alta		
Verificabilidad: Alta		

28 Requisito Restricción UR-0025

ID: UR – 0025	Tipo: Restricción	Necesidad: Media
Título: Seguridad de sesión		
Fuente: Cliente		
Descripción: La sesión del usuario tendrá un tiempo de caducidad.		
Estabilidad: Media		
Verificabilidad: Alta		

29 Requisito Restricción UR-0026

ID: UR – 0026	Tipo: Restricción	Necesidad: Alta
Título: Pedidos En Caja		
Fuente: Cliente		
Descripción: Las cajas a enviar contendrán cómo mínimo un pedido en ellas.		
Estabilidad: Media		
Verificabilidad: Alta		

30 Requisito Restricción UR-0027

ID: UR – 0027	Tipo: Restricción	Necesidad: Alta
Título: Cajas En Cargamentos		
Fuente: Cliente		
Descripción: Los cargamentos a enviar tendrán como mínimo una caja cerrada en él.		
Estabilidad: Media		
Verificabilidad: Alta		

31 Requisito Restricción UR-0028

ID: UR – 0028	Tipo: Restricción	Necesidad: Alta
Título: Cargamentos En Envíos		
Fuente: Cliente		
Descripción: Los envíos tendrán cómo mínimo un cargamento cerrado en ellos.		
Estabilidad: Media		
Verificabilidad: Alta		

6.2. Requisitos Técnicos

A continuación se muestran los requisitos técnicos obtenidos.

32 Requisito Técnico Rq_Tec_1

ID: Rq_Tec_1	Tipo: Sistema	Necesidad: Alta
Título: Característica de la Base De Datos		
Fuente: Cliente		
Descripción: La base de datos deberá ser consistente.		
Estabilidad: Alta		
Verificabilidad: Alta		

33 Requisito Técnico Rq_Tec_2

ID: Rq_Tec_2	Tipo: Sistema	Necesidad: Alta
Título: Seguridad de la Base De Datos		
Fuente: Cliente		
Descripción: Seguridad (información asegurada ante usuarios) (niveles y control de acceso).		
Estabilidad: Alta		
Verificabilidad: Alta		

34 Requisito Técnico Rq_Tec_3

ID: Rq_Tec_3	Tipo: Sistema	Necesidad: Alta
Título: Integridad de la Base De Datos		
Fuente: Cliente		
Descripción: Los datos almacenados en la base de datos deberán ser válidos.		
Estabilidad: Alta		
Verificabilidad: Alta		

35 Requisito Técnico Rq_Tec_4

ID: Rq_Tec_4	Tipo: Sistema	Necesidad: Alta
Título: Consulta de la Base De Datos		
Fuente: Cliente		
Descripción: La base de datos contará con respaldo y recuperación de líneas con alta eficiencia.		
Estabilidad: Alta		
Verificabilidad: Alta		

36 Requisito Técnico Rq_Tec_5

ID: Rq_Tec_5	Tipo: Sistema	Necesidad: Alta
Título: Tiempo de respuesta de la Base De Datos		
Fuente: Cliente		
Descripción: El tiempo de respuesta será mínimo.		
Estabilidad: Alta		
Verificabilidad: Alta		

7. Análisis del sistema

Una vez obtenidos los requisitos de usuario será necesario realizar la especificación funcional de la herramienta a desarrollar que verifique el cumplimiento de los requisitos del sistema y satisfaga las necesidades del usuario, siendo compatible con el resto de los subsistemas que deba interactuar.

7.1 Elaboración del modelo de procesos

7.1.1 Obtención del Modelo de Procesos del Sistema

Introducción. A continuación se describe el sistema generalizado para realizar pedidos vía estándar FMS a Estados Unidos. Inicialmente se reflejará todo el proceso con su intercambio de documentación entre el cliente con EE.UU e iremos descendiendo hasta llegar a nuestro núcleo que va a desempeñar nuestro gestor de almacén pasando por la propia gestión de la mercancía.

El proceso se engloba en 4 bloques, los cuales serán explotados en detalle para un mejor entendimiento.

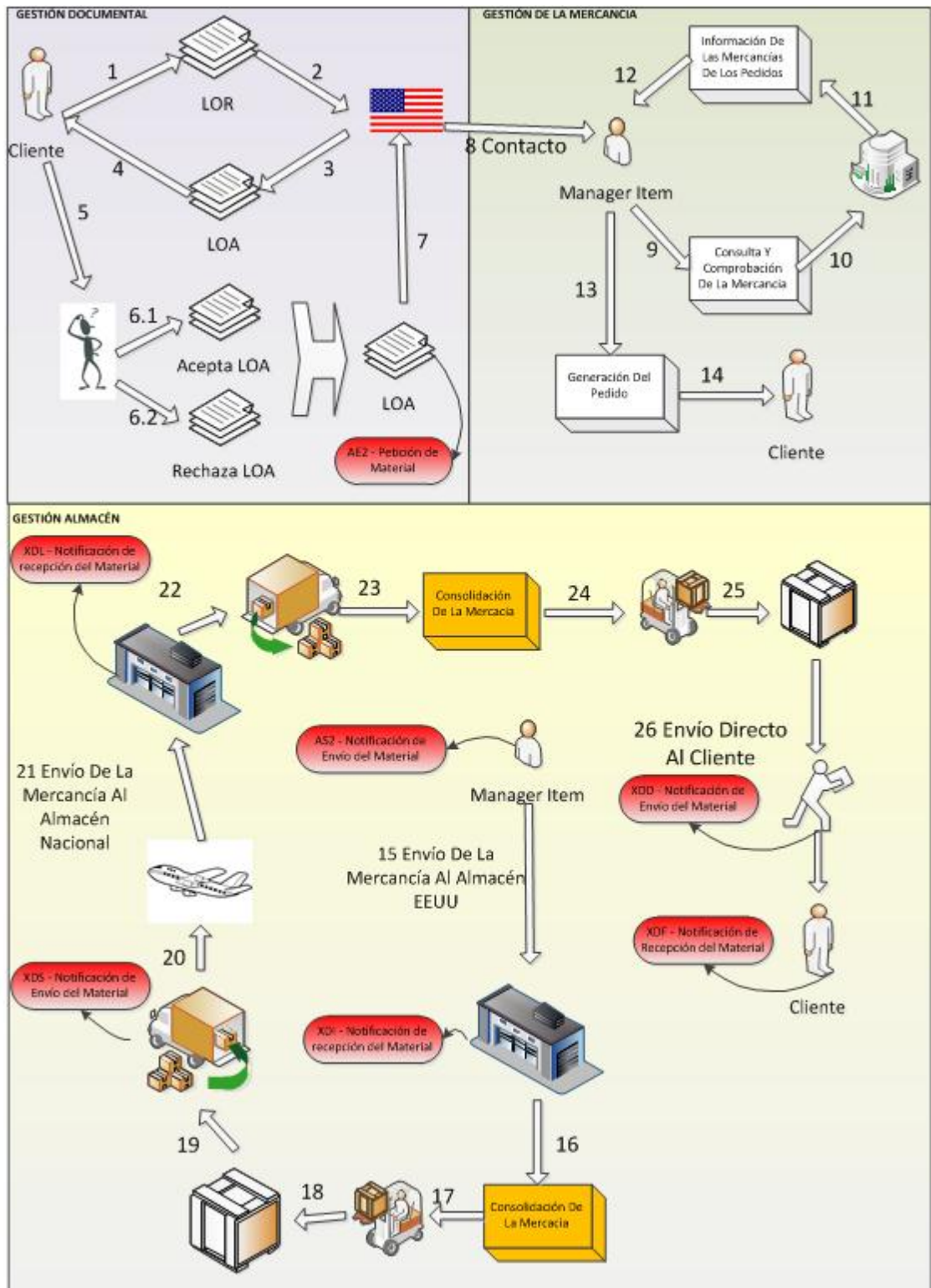
Los 4 bloques son:

1. **Gestión Documental:** Describe el proceso de solicitud de permisos para el envío de un pedido de una mercancía de nueva adquisición o petición de reparable.
2. **Gestión De La Mercancía:** Es el proceso que tienen que realizar los ejércitos de todo el mundo para la compra de material, el cual, se puede considerar como la preparación del pedido.
3. **Gestión De Almacén de los pedidos.** Es el proceso que se realiza en los almacenes a partir de la entrada de la notificación de solicitud de una mercancía.
4. Y por último la **integración de los procesos** descritos anteriormente con el sistema de información global FMS.

Estos cuatro bloques a su vez sufren una pequeña variación en su gestión en función de si se trata de una mercancía que llega al sistema como nueva adquisición o se trata de un material reparable. Mayormente el proceso se comporta del mismo modo para ambos casos, salvo que cuando se trata de un material reparable el proceso incluye la recogida del material en el cliente y el envío de este a la fuente de reparación, pero una vez que la mercancía se encuentra en dicha fuente de reparación y el material ha sido reparado, el proceso de vuelta es el mismo que de una mercancía de nueva adquisición.

A continuación se muestra el diagrama del proceso completo de un pedido de mercancía de nueva adquisición a EE.UU:

Nueva Adquisición.



9 Pedido Nueva Adquisición

Por tanto las funcionalidades que debe cumplir el Sistema con el estándar se resumen en la existencia de las líneas FMS (AS2, XDI, XDS...), en cada una de las gestiones pertinentes.

Nota: Tal como se ha comentado anteriormente, dicho sistema sólo reflejará la consolidación de la mercancía en el almacén junto con su creación de líneas FMS(Cuadros naranjas de la imagen).

A continuación se van a detallar dichas gestiones recogiendo el estándar y trazabilidad de líneas.

Gestión Documental

- **1.** El cliente entrega el documento LOR a EEUU. En dicho documento establece el conjunto de materiales que quiere adquirir.

Las mercancías que entran en dicha solicitud son:

- Elementos sueltos · Partes de recambio · componentes menores.

Mercancías que no forman parte de dicha solicitud:

- El material clasificado y publicaciones no autorizadas.
 - Servicios de soporte técnico, incluyendo servicios de control de calidad, inspecciones, y revisiones de cuentas
 - Piezas de recambio y reparables que necesiten un apoyo especial.
 - Armamento y artículos explosivos, incluyendo torpedos, minas, ojivas, y todas las armas de fuego y otras armas portátiles.
 - Componentes característicos, incluyendo sistemas propios aeroespaciales y marítimos. (ejemplo misiles, radares y sónar)
- **2.** EEUU estudia la LOR recibida y se dispone a comprobar las mercancías que el cliente quiere solicitar. Si todo está correcto contacta con las áreas de control encargadas de dichos materiales y genera la **LOA**. Si EEUU necesita una información extra por parte del cliente, le será comunicado al cliente antes de generar la LOA, durante, o en el mismo proceso o gestión de la mercancía.
 - Si el cliente no responde en un plazo fijado a cerca de la información extra que necesite EE.UU, el proceso queda totalmente cancelado.
 - **3.** Una vez que llega la LOA al cliente, éste la estudia y puede:

- Aceptarla, enviando dicha aceptación con la información necesaria de financiación, transporte e información.
- Rechazarla. Si es rechazada, el cliente puede volver a contactar con EEUU enviando una nueva LOR volviendo al paso 1.

Si el cliente ha aceptado la LOA. Una vez que es recibida en EEUU, se procede a consultar el stock de cada mercancía y su generación de pedido, por lo que entramos en el siguiente bloque.

Gestión De La Mercancía

- 1. Una vez que se tiene toda la información de la mercancía solicitada, se contactar con los “Manager Item”, los cuales son los encargados de gestionar dicha mercancía con su número de stock, localización e incluso si se sigue fabricando o si es una mercancía clasificada en su forma de tratar. Ellos son los encargados de contactar con los propios proveedores de suministros del ejército y su solicitud de compra.
- 2. Los Manager Item se encargan de comprobar las mercancías solicitadas y si no se produce ninguna discrepancia se genera el pedido de la mercancía.
- 3. En el caso que no haya discrepancia y que la mercancía pueda ser servida, es el Manager Item es el encargado de la generación del documento de aviso electrónico del envío de la misma. (Generación del AS2)

Gestión De Almacenes

- 1. Una vez que llega la mercancía al almacén de EE.UU (generación del XDI) es donde entra en funcionamiento nuestro gestor de almacén.
- 2. Se lleva a cabo el proceso de consolidación para el posterior envío al almacén del cliente que solicitó la mercancía. En dicha consolidación se realiza una:
 - **Gestión de cargamentos:** Los cargamentos son los contenedores más grandes en los cuales irán las cajas de las que se componen los pedidos. En dicha gestión es posible crear un nuevo cargamento en el propio almacén. Cerrar un cargamento para su posterior envío y abrir un cargamento para poder obtener las cajas guardadas en su interior.

- **Gestión de cajas.** Las cajas tienen menor tamaño que los contenedores en las cuales irán las mercancías de un pedido. Es posible crear nuevas cajas desde el almacén, abrirlas para introducir nuevos pedidos y cerrarlas para poder ser introducidas posteriormente en un contenedor.

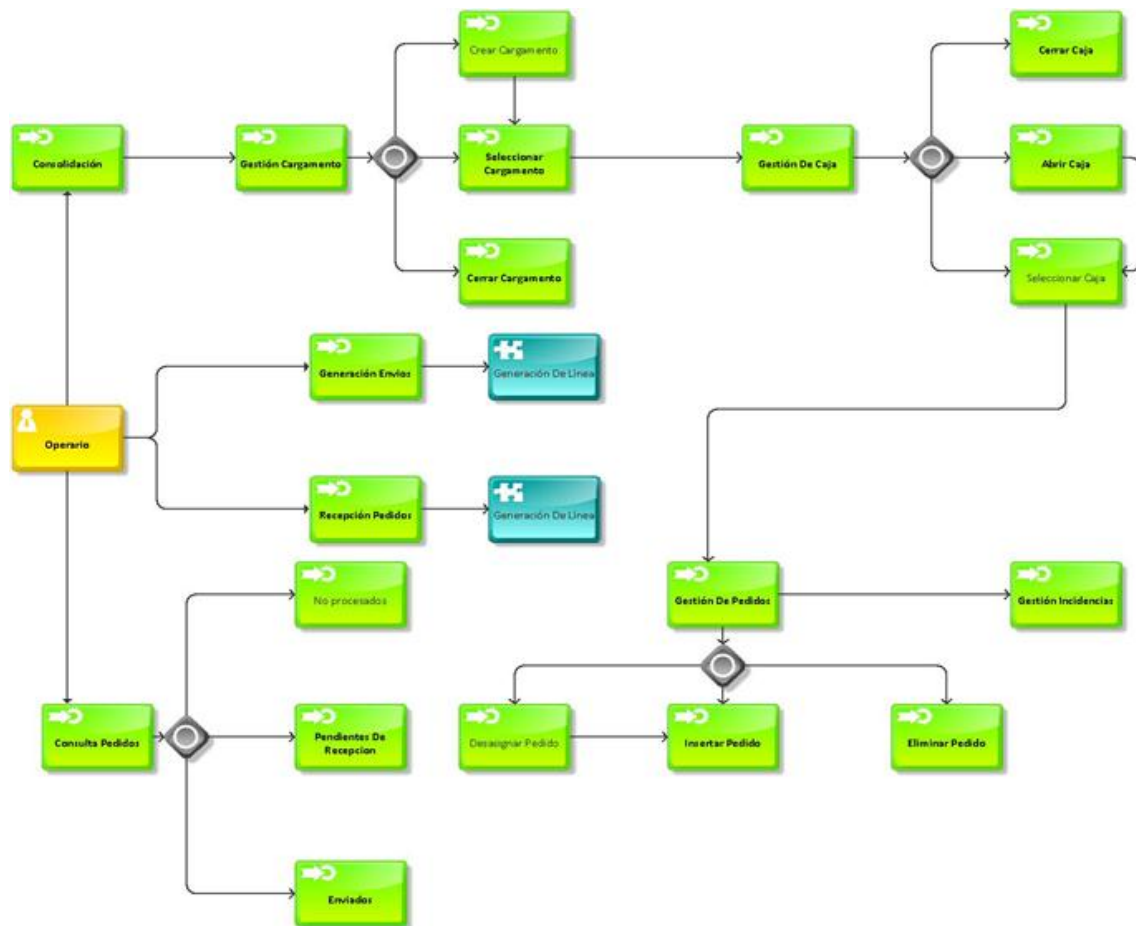
- **Gestión de pedidos.** Un pedido es una mercancía lista para enviar al almacén del cliente comprador. Un pedido siempre debe ser embalado dentro de una caja. En caso de ser un pedido compuesto por varias puestas, éste será embalado en el número de cajas e incluso de cargamentos necesarios para su envío. Es posible desasignar pedidos de las cajas para poder asignarlos a otras cajas e incluso eliminar dicho pedido del almacén por alguna causa excepcional.

- **Consulta de pedidos.** Dentro del gestor es posible consultar los pedidos para analizar o tomar datos relevantes de ellos pudiendo consultar pedidos directamente, pedidos modificados, sin asignar o pendientes de recepción.

A continuación se muestra un diagrama de procesos más técnico centrado puramente en la gestión del almacén.

Lógica de procesos interno del almacén

Para el correcto funcionamiento de la aplicación será necesario definir y describir la lógica de los procesos que interactúan y su secuenciación comprobando su funcionamiento.



10 Lógica De Procesos De Almacén

La aplicación de almacén se basa en 4 procesos generales que son la consolidación, proceso de consultas, generación de envíos y recepción de envíos.

Las gestión de cargamento incluye el crear un nuevo cargamento en el almacén, posibilitar la apertura de dicho cargamento para introducir nuevas cajas, y el poder cerrar el cargamento y dejarlo listo para enviar.

La gestión de caja es similar al de cargamento, sólo que en las cajas van incluidas los pedidos. Se podrá crear una nueva caja, cerrarla o abrirla para insertar o modificar su contenido.

A nivel de caja es donde se introducen los pedidos, para introducir un pedido se inserta el pedido en caja. A este nivel también es posible eliminar el pedido por causas justificadas y el poder desasignarlo de una caja para poder introducirlo en otra a causa de error o cambio en la documentación inicial entre el cliente y el Manager Item.

El último núcleo de funcionalidad del almacén son las consultas:

- Pedidos pendientes de recepción. Todavía no han llegado al almacén pero se está a la espera. Se tiene el AS2 pero todavía no el XDI generado.
- Pedidos no procesados. Todavía no han sido consolidados y embalados para su posterior envío.
- Pedidos enviados. Han sido consolidados y enviados al cliente.

El proceso de generación de envíos se desarrolla cuando el operario agrupa cargamentos cerrados para su proceso de envío al cliente. Se generan las líneas XDS

El proceso de recepción de envíos se realiza cuando el operario con su pistola, lee el código de barras de las mercancías llegadas al almacén. Se generan las líneas XDI

Tenemos los dos roles fijados:

- Operario del almacén: puede gestionar toda la aplicación.
- Cliente: sólo le está permitido entrar al nivel de consultas de pedidos generales, para conocer la situación actual de su pedido.

Integración con el sistema de información global de trazabilidad.

Aunque a lo largo de la descripción de los procesos anteriores ya se ha introducido brevemente cómo se integran algunos de los pasos de estos procesos con el sistema de información FMS, a continuación, de se va a detallar completamente como se realiza dicha integración.

Se ha dividido el proceso en 2 partes

1. Gestión de Procesos en el extranjero.
2. Gestión de Procesos Nacional.

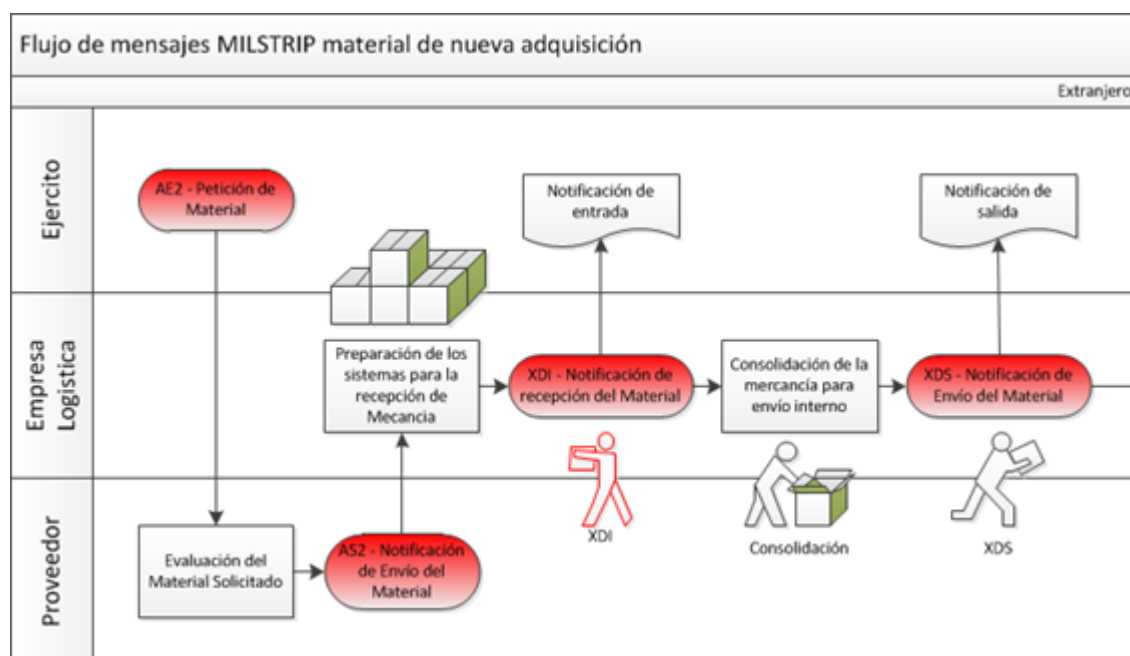
Realmente hay una parte más en la que está dividido el sistema, que es la parte de la gestión de la mercancía reparable. La cual se explicará en dicho apartado.

Gestión del proceso en el extranjero.

Cada una de las transacciones que se realizan con la mercancía tiene que pasar por el sistema de información de trazabilidad de pedidos para que este comunique a todos los interlocutores el movimiento que se está realizando con dicha mercancía.

Hay que explicar es que dentro de la nomenclatura “Empresa Logística” se han englobado el sistema de información logístico global para la trazabilidad de los pedidos y el sistema de gestión de almacén, debido a que el sistema de información de trazabilidad, alimenta y recoge información del sistema de gestión de almacén, a la vez que también se comunica con los sistemas logísticos militares de los gobiernos de que realizan las solicitudes de compra de material. (Ver Comunicación entre sistemas)

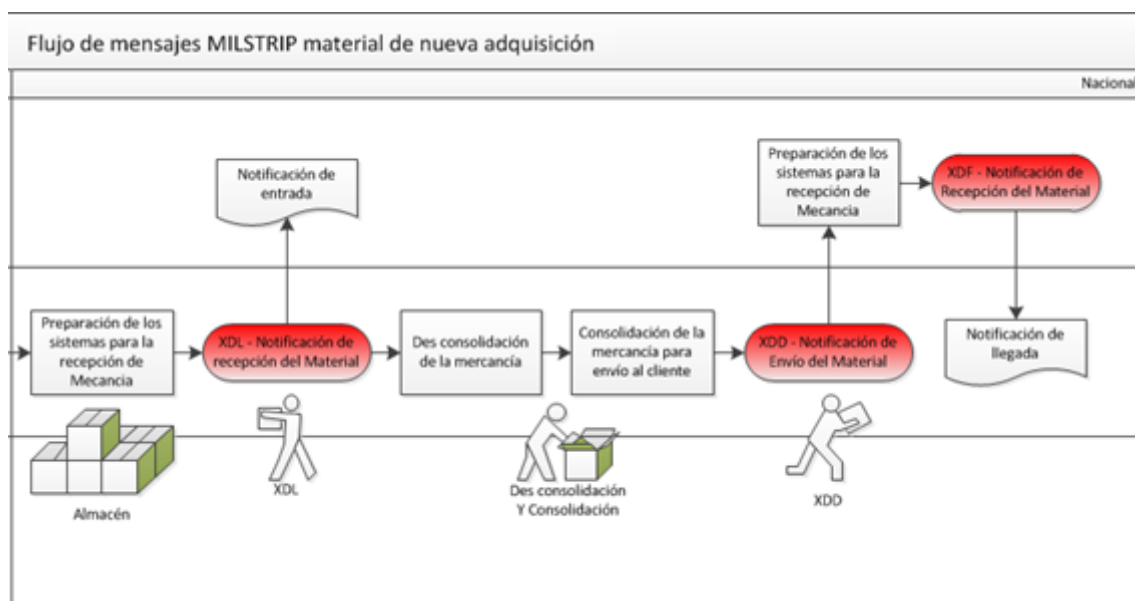
11 Flujo Material Nueva Adquisición Extranjero



1. Es sistema de información para la trazabilidad de los pedidos FMS, de ahora en adelante **SYSTRACK-FMS**, es el que se encarga de recoger del DAAS las solicitudes que los ejércitos realizan de materiales. Estas solicitudes son las que llevan el formato del **MILSTRIP-AE2**.
2. Justo cuando el sistema detecta que el proveedor del material va a realizar el envío de la mercancía, (**MILSTRIP-AS2**), este se encarga de avisar al sistema de gestión de almacén de la futura entrada de un material, con el fin de que vayan preparando la ubicación donde va a ser emplazado dicho material. Esto se debe a que existen materiales de grandes dimensiones.
3. El sistema de gestión de almacén una vez que ha recibido la mercancía en sus instalaciones ha de comunicar a SYSTRACK-FMS que dicha recepción se ha realizado correctamente a través de una línea (**MILSTRIP-XDI**).
4. Cuando el almacén ha consolidado un cargamento, esto es que ha llenado un contenedor con parte de la mercancía que tiene pendiente de enviar, y por lo tanto ya se dispone a realizar el envío desde sus almacenes en EEUU hacia el país de destino, ha de notificar dicha salida de la mercancía a través de una línea (**MILSTRIP-XDS**).

Gestión del proceso nacional.

Aunque no es necesario el SYSTRACK-FMS para la comunicación de la información logística entre almacenes, puesto que es una información que está recogida en el sistema de información para la gestión de los almacenes, SYSTRACK-FMS es necesario para comunicar al DAAS que dicho movimiento de mercancía ya se ha producido.



1. Al igual que sucede cuando la mercancía es recibida del proveedor, el sistema de gestión del almacén tiene que notificar a través del **MILSTRIP-XDL** al SYSTRACK-FMS que la mercancía ha llegado al país de destino.
2. Físicamente en el almacén se realizará la manipulación necesaria para preparar la mercancía recibida, preparándola para su envío al peticionario. El sistema de gestión de almacén ha de notificar al SYSTRACK-FMS dicho movimiento con un **MILSTRIP-XDD**.
3. Finalmente cuando el peticionario recibe la mercancía en sus instalaciones tiene que generar un **MILSTRIP-XDF** indicando dicha recepción.

Pedido Reparable.

Si el cliente quiere mandar a EE.UU una mercancía a reparar, el proceso será similar pero con algunas peculiaridades en los distintos bloques.

Gestión Documental Reparable

El proceso es similar al analizado anteriormente.

La diferencia es en el envío de la documentación necesaria de la mercancía a reparar.

En el LOR debe incluirse todos los detalles de la mercancía a reparar que va a ser enviada. El envío aunque pueda ser conjuntamente, cada mercancía debe ser gestionada de forma única.

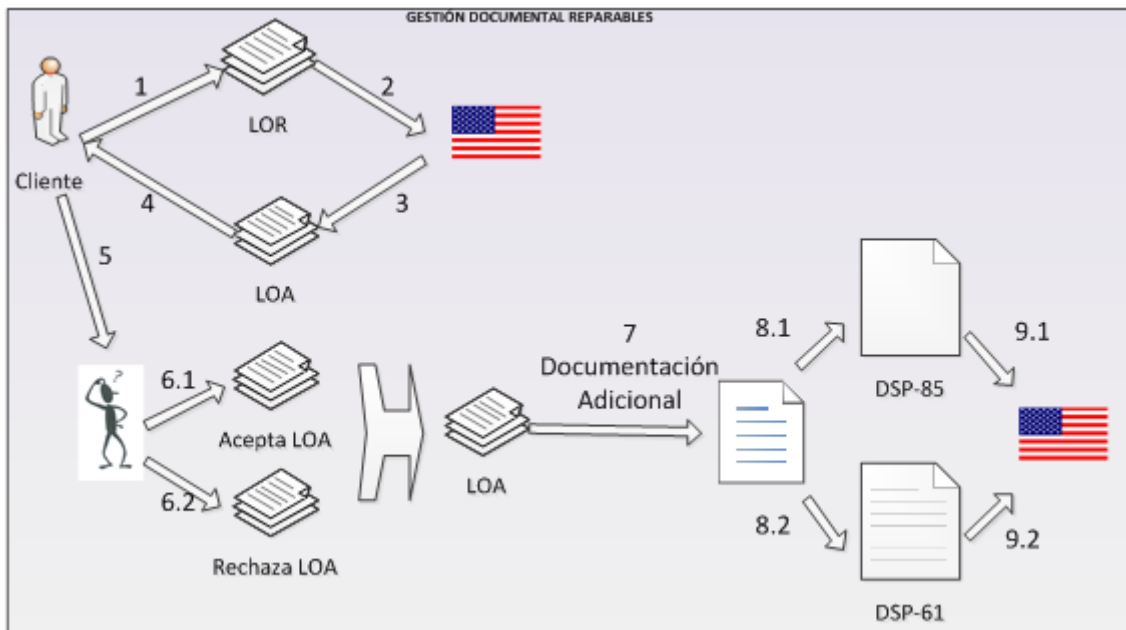
Como en el caso de mercancías normales, se estudia la LOR y se genera la LOA estableciendo la información necesaria que debe enviar el cliente para dichos reparables.

El cliente en caso de aceptar la LOA enviará una documentación adicional:

Al enviar un artículo, éste debe pasar obligatoriamente por la aduana de EEUU. Para ello es necesario que el cliente cumplimente unos documentos para identificar las mercancías y sus destinos para llevar a cabo las reparaciones.

- Forma DSP-61 Para artículos no clasificados.
- Forma DSP-85 para artículos especiales o confidenciales.

Nota: no es posible enviar un conjunto de elementos para su reparación, su envío será de uno en uno.



13 Gestión Documental Reparable

Antes de enviar cualquier mercancía, se debe comprobar que es posible su reparación y establecer unos costes y fechas de entrega.

Para ello entramos en el siguiente bloque de gestión de la mercancía de reparables.

Gestión De La Mercancía De Un Reparable

Tenemos un nuevo proceso que es la comprobación del TRIL.

Se debe saber si dicho elemento detallado en la documentación por parte del cliente (formas DSP) forma parte del TRIL.

El TRIL especifica los artículos reparables de cada sistema u organismo y sus instalaciones de reparación correspondientes, ya que es posible que una misma mercancía deba ser reparada en dos o más talleres.

Al comprobar el TRIL tenemos tres opciones:

- La mercancía forma parte del TRIL por lo que no se tiene ningún problema y se inicia la posterior gestión.
- La mercancía no forma parte del TRIL y se procede al estudio de poder incluirla para realizar la posterior reparación. El no formar parte del TRIL puede deber por varias razones como que el reparable ya está en desuso y se ha descatalogado o directamente se ha dejado de dar soporte a dicha mercancía. Dicho proceso puede incrementar muchísimo el tiempo de espera por lo que es notificado al cliente por si éste no dispone de dicho tiempo.

Toda mercancía debe estar incluida en dicha documentación obligatoriamente para ser reparada.

- La mercancía no forma parte del TRIL y se decide no incluirla. Por lo tanto se rompe el procedimiento al no poder repararse la mercancía requerida.

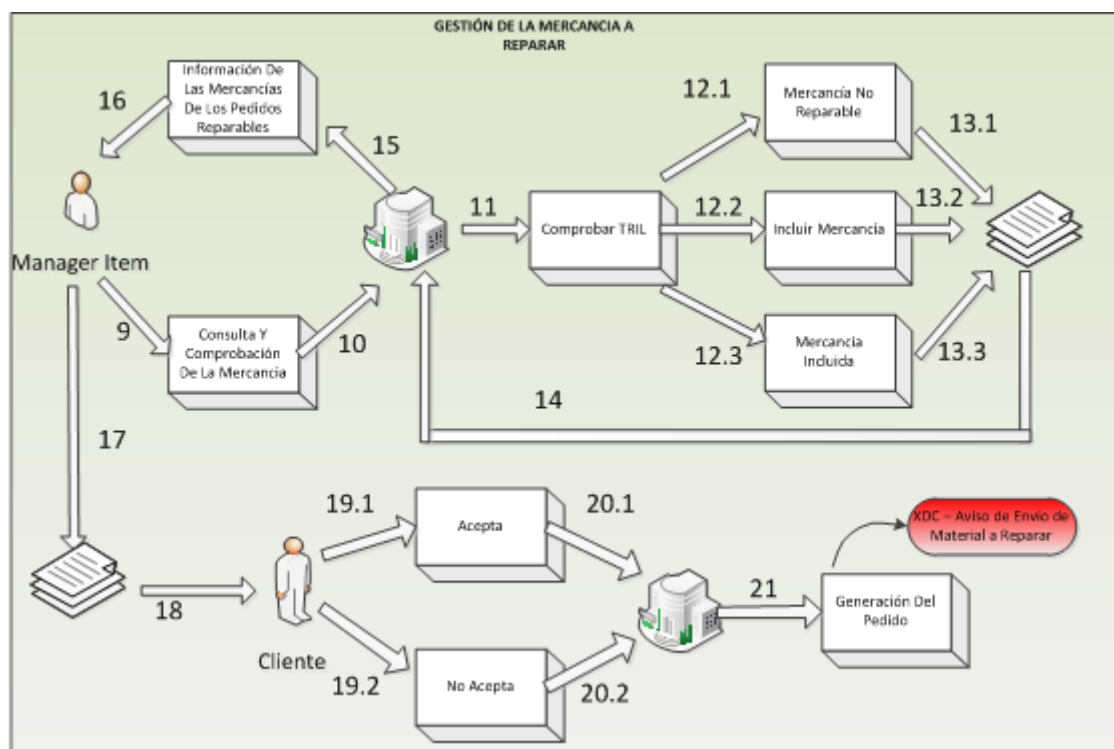
Una vez que se sabe que la mercancía forma parte del TRIL, se procede a obtener toda la documentación necesaria para la propia reparación cómo donde va a ser reparada, el coste, los plazos y las posibles discrepancias que se puedan originar.

Se prepara un último documento en el que se refleja una estimación de costos para los servicios de reparación para cada artículo. Se identifica el artículo que va a ser reparado especificando el grado de reparación.

Si el cliente acepta la reparación nos introducimos en el siguiente bloque en el cual se deberá embalar la mercancía y enviarla a los almacenes de EE.UU.

Si el cliente rechaza la reparación, puede volver a repetir dicho proceso desde la entrega de la documentación.

En el siguiente esquema se refleja dicho proceso:



Gestión De Almacén De Un Reparable

Desde el almacén o la propia industria del cliente se debe enviar la mercancía al almacén de EE.UU.

Con los documentos oficiales, el cliente deberá embalar la mercancía tal como se ha estipulado.

Una vez que ha llegado, se comprueba que la mercancía corresponde con lo estipulado y se lleva a cabo una revisión para comprobarla.

Proceso llevado a cabo cuando llegar el artículo reparable a EEUU:

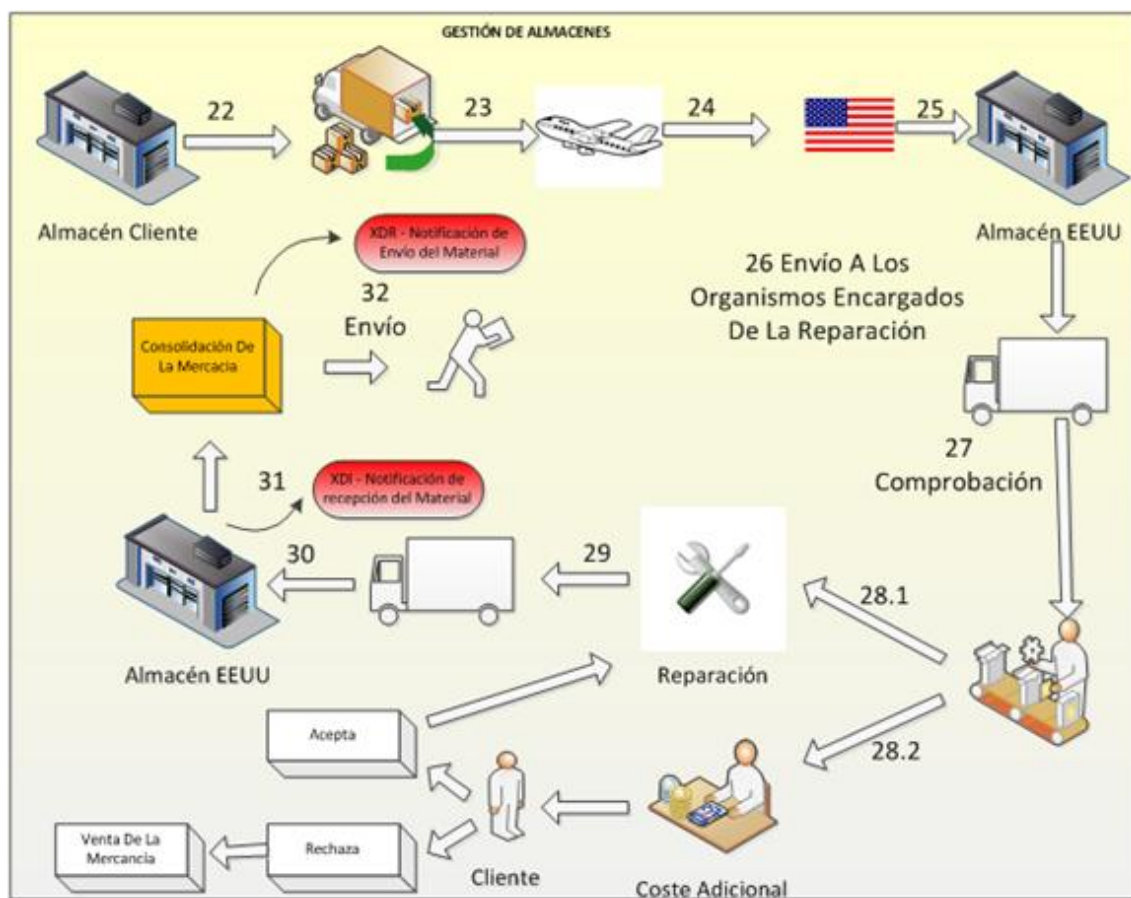
- Inspeccionan el artículo para comprobar la exactitud y la información proporcionada en documentación.
- Comprobar si los artículos se sitúan dentro de la reparación el cliente ha informado anteriormente mediante una inspección técnica.
- Si el coste para reparar el artículo excederá el precio corriente estándar:
 - Se informará al cliente que el coste para completar la reparación excederá el precio corriente estándar.
- El cliente puede responder:
 - Autorizan la financiación adicional para seguir con el trabajo de reparación. Si el cliente acepta las nuevas modificaciones del contrato, se llevará a cabo la reparación del producto en las localizaciones estipuladas.
 - Deseo de devolución del artículo no reparado al cliente.
 - Si el cliente reniega de la reparación se origina un nuevo proceso en el cual el cliente da plenos poderes para que se produzca una venta de dicha mercancía. Un porcentaje obtenido de la venta de la mercancía irá a parar hacia el cliente mientras que otro porcentaje será ingresado por el departamento propio que ha llevado dicha gestión.

Una vez que la mercancía ha sido reparada y llevada al almacén propio de EEUU (paso 30) , es donde se realiza la consolidación de dicha mercancía para su posterior envío al cliente.

Existen distintos envíos por parte del cliente y EEUU:

- Envío Asegurado. Los clientes pueden asegurar contra la pérdida (de o dañar 5-18 a) artículos reparables mientras ellos están en el tránsito. Tal seguro puede ser obtenido de una firma comercial de seguros.
- Auto seguro. Los clientes simplemente pueden asumir el riesgo ellos mismos, comprendiendo que si un artículo es perdido, ellos tendrán que sustituirlo en su propio costo.

Los artículos reparables no son perdidos frecuentemente en el envío. Sin embargo, aconsejan que clientes busquen un seguro para una buena protección en caso de pérdida.

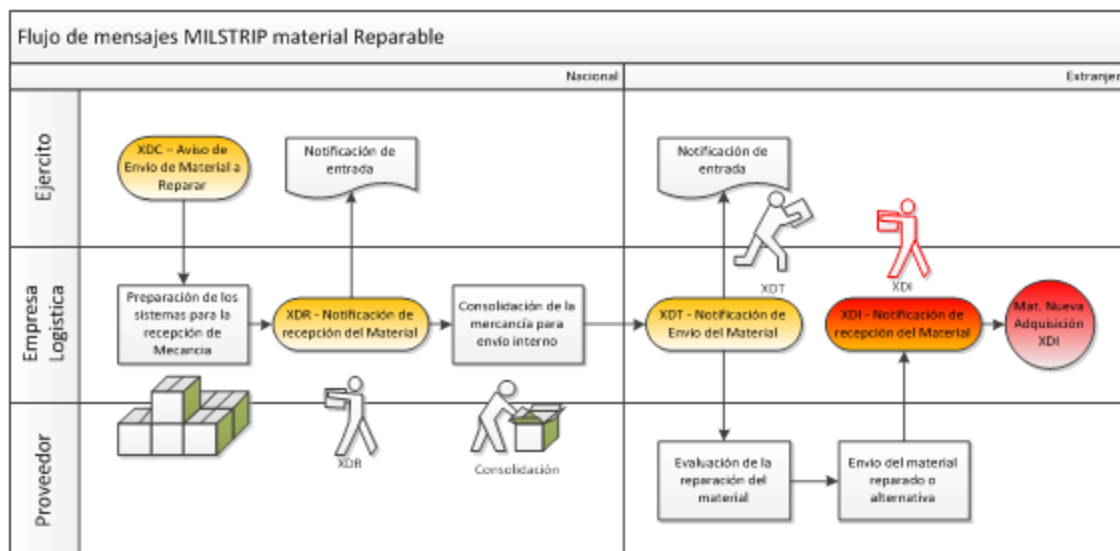


15 Gestión Almacenes Reparable

El seguro puede parecer a un costo innecesario. Sin embargo, debido al generalmente alto costo de artículos reparables, la pérdida no asegurada de solamente un artículo equivale a una gran pérdida económica por parte del cliente.

La consolidación es la misma que la realizada en el caso de mercancías no reparables.

Integración con el sistema de información global de trazabilidad.

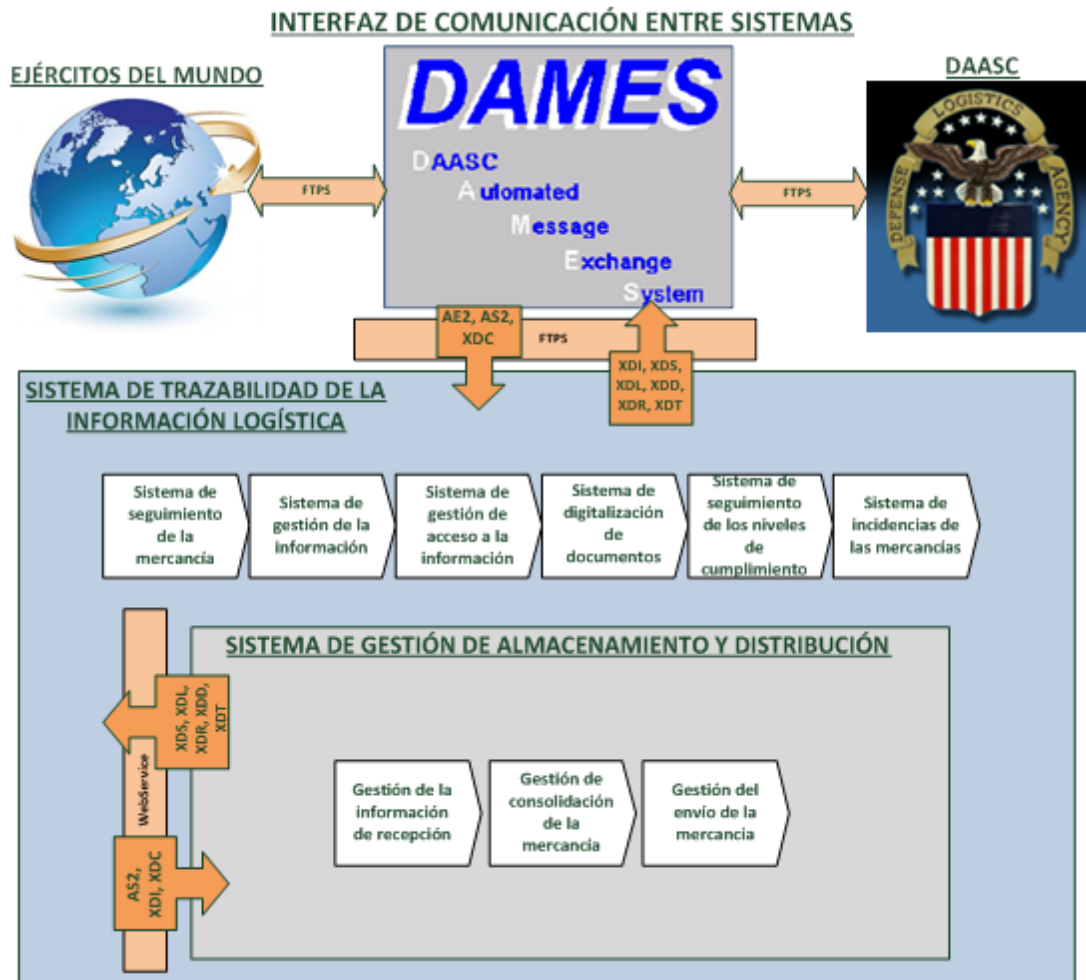


16 Flujo Material Reparable

1. Cuando algún ejército requiera la reparación de algún material previamente adquirido, deberá informar al sistema logístico global a través de la generación de un **MILSTRIP-XDC** de que quiere enviar dicho material a la fuente de reparación. Y será el SYSTRACK-FMS el encargado de recoger esta petición e informar al almacén la inmediata llegada de dicho material a los almacenes.
2. Al igual que en el proceso de nueva adquisición, cada vez que la mercancía es recibida en el almacén, es el sistema de gestión de almacén el encargado de notificar al SYSTRACK-FMS a través de la generación del **MILSTRIP-XDR** la entrada de la mercancía en sus instalaciones.
3. Y siguiendo con el paralelismo al proceso de nueva adquisición, el sistema de gestión de almacén generará un **MILSTRIP-XDT** para que el SYSTRACK-FMS notifique al sistema global de trazabilidad la salida de este material hacia el país destino en el cual se va a realizar la reparación.
4. Una vez que el material ha sido evaluado para su reparación o sustitución, la fuente de reparación envía físicamente al almacén el material ya reparado o el sustituto. Con lo cual el sistema de gestión de almacén generaría el **MILSTRIP-XDI** para indicar la recepción de dicho material. El proceso en este punto enlazaría con el proceso de material de nueva adquisición.

7.1.2 Especificación de Interfaces con otros Sistemas

El sistema debe interactuar con otros subsistemas que integren el proceso de la logística total y para ello se definirán las interfaces y los protocolos de comunicación con ellos y que se describen a continuación



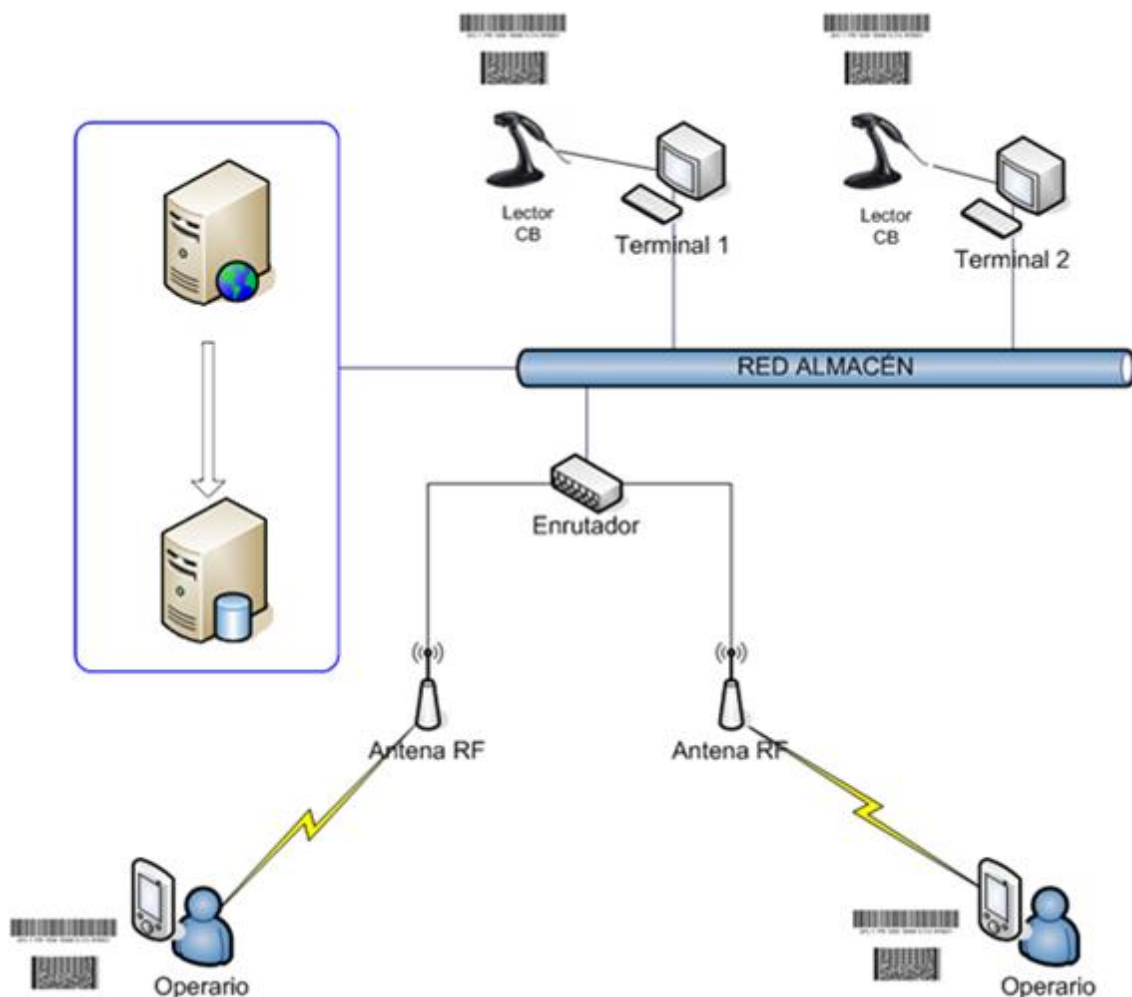
El sistema de comunicación consta de dos partes claramente diferenciables:

1. La primera es la que establece las comunicaciones del sistema de trazabilidad con los Sistemas de Información de los ministerios de defensa con los fabricantes y almacenes, la cual se realiza de acuerdo con las estipulaciones de adquisición de **material FMS y comercial**, a través de la **Red Logística de Comunicaciones Internacionales (I.L.C.S.)**.

La comunicación entre el sistema de trazabilidad y los sistemas de información de los ministerios, se realiza a través de una aplicación designada por el DAAS, llamada DAMES, la cual es común para todos los organismos que realizan compras por medio FMS. El protocolo de comunicación requerido es el

protocolo FTPS. En la cual cada organismo tiene designado un usuario y contraseña que es la que le da acceso a un directorio en que se envía la información generada por el sistema y se recoge toda la información que el resto de los sistemas de los otros organismos también han generado.

2. La segunda es la que se utiliza para comunicar el sistema de trazabilidad de la mercancía con el sistema de gestión de almacén. Al ser ambas aplicaciones parte de un mismo sistema, la comunicación entre ambas se realiza a través de servicios web, sobre el protocolo http, puesto que no es necesario el encriptado para este tipo de comunicación, al estar dentro de una red segura o intranet.



7.2 Análisis de los casos de uso

7.2.2 Funcionalidades del sistema.

Antes de definir el diagrama de casos de uso, vamos a enumerar las funcionalidades de nuestro gestor de almacén.

Las funcionalidades se dividen en 3 bloques:

- Consultas: para obtener información de los pedidos, mercancías y embalajes tanto para el operario de almacén como para el cliente. El cliente también podrá acceder a dichas consultas.
- Consolidación: es el núcleo de la aplicación, donde se realizan las funciones de embalado e inserción de las mercancías en cajas o contenedores.
- Gestión De Incidencias: se podrán dar de alta de incidencias en los posibles errores o defectos producidos en el transcurso de la consolidación. Éste bloque en sí, pertenece al proceso de consolidación, pero para tener una mejor visualización de los casos de uso, se ha reflejado una funcionalidad fuera de dicho proceso.

Dentro del bloque de *Consultas* se reflejará:

- Pedidos pendientes de recepción: son los pedidos que se esperan en el almacén. Equivalen a los pedidos que contienen la línea AS2 y no la XDI.
- Pedidos enviados: los cuales ya han sido consolidados y se ha generado su línea XDS.
- No procesados: todavía no han sido consolidados y esperan en el almacén. Tienen su línea XDI pero no su XDS.

Dentro del bloque de *Recepciones* se reflejará:

- Recepción de pedidos: cuando llegan diariamente los pedidos al almacén, el operario deberá recepcionarlos para así crear su XDI y poder consolidarlos a su país solicitante.
- Gestión de caja: el operario introducirá los pedidos en cajas, para ello podrá:
 - Abrir una caja existente.
 - Cerrar caja con sus pedidos ya introducidos.
 - Crear nueva caja.

- Gestión de contenedores: el operario introducirá las cajas en contenedores, para ello podrá:
 - Abrir contenedor existente.
 - Cerrar contenedor con sus cajas ya introducidas.
 - Crear nuevo contenedor.
- La Consolidación. Es el núcleo del sistema en el cual se verificarán las mercancías y embalajes para su embalaje en cajas, cargamentos y contenedores:
El operario deberá:
 - Seleccionar o crear un nuevo Cargamento.
 - Seleccionar o crear una nueva caja para ese Cargamento.
 - Insertar el pedido en dicha caja o cajas.
 - Cerrar la caja y el cargamento para finalizar la consolidación.
- Envío: se procederá a seleccionar los cargamentos que conformarán dicho envío. Una vez que es cerrado el envío son generadas las líneas XDS de los pedidos incluidos en él.

Dentro del bloque de *Incidencias* se podrán crear incidencias originadas en el proceso de consolidación.

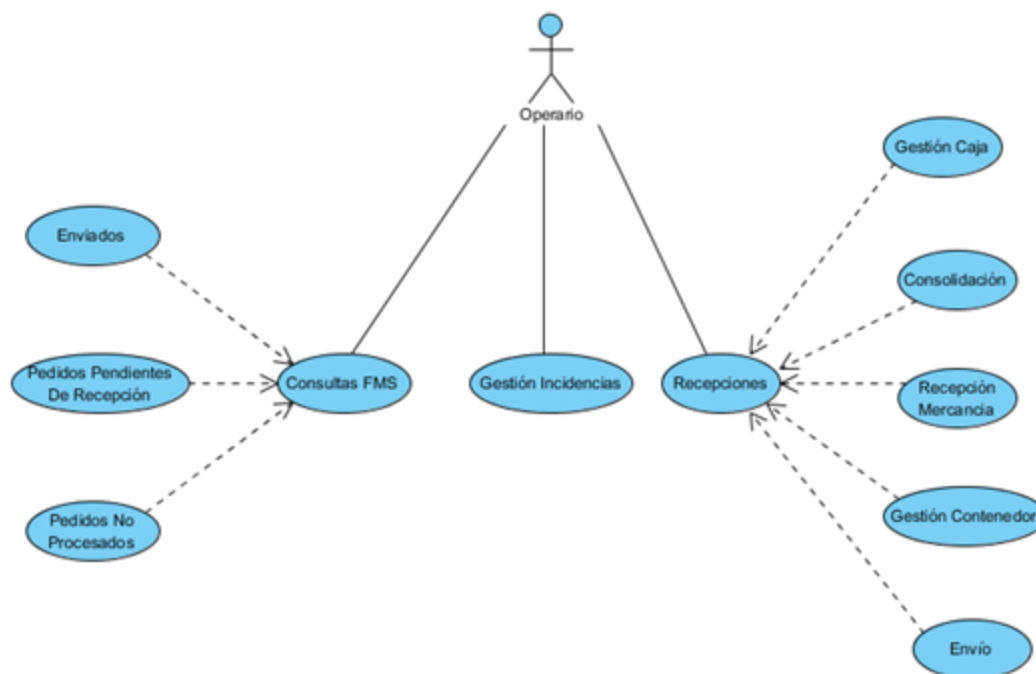
7.2.2.1 Definición de stakeholders.

En el sistema van a existir 2 tipos de usuarios.

- Administrador: Este tipo de usuario es el encargado en dar de altas nuevos usuarios en la aplicación, habilitarlos, deshabilitarlos y tiene la posibilidad de dar de alta nuevos administradores.
- Operario: Esta cuenta pertenecerá a aquella persona experimentada y capaz de realizar la consolidación de los pedidos del almacén. Podrá realizar cualquier gestión dentro de la aplicación.

7.2.3 Especificación de Casos de Uso

7.2.3.1 Operario



19 Diagrama De Casos De Uso Operario

Definición De Los Casos De Uso:

Generales:

37 Caso De Uso Consultas FMS

Caso de Uso:	Consultas FMS
Actores:	Operario
Objetivo:	Obtener los datos de los pedidos para poder operar con ellas.
Precondiciones:	El pedido debe contener su línea XDI

38 Caso De Uso Gestión De Incidencias

Caso de Uso:	Gestión De Incidencias
Actores:	Operario
Objetivo:	Dar de altas incidencias a los pedidos procesados en la consolidación.
Precondiciones:	El pedido debe contener su línea XDI

39 Caso De Uso Recepciones

Caso de Uso:	Recepciones
Actores:	Operario
Objetivo:	Realizar la consolidación del pedido solicitado.
Precondiciones:	El pedido debe contener su línea AS2

Extendidos:

De consultas FMS:

40 Caso De Uso Pedidos Enviados

Caso de Uso:	Pedidos enviados
Actores:	Operario
Objetivo:	Obtener los datos de los pedidos enviados
Precondiciones:	El pedido debe contener su línea XDS

41 Caso De Uso Pedidos Pendientes

Caso de Uso:	Pedidos pendientes de recepción
Actores:	Operario
Objetivo:	Obtener los datos de los pedidos que no han sido recepcionados en el almacén.
Precondiciones:	El pedido debe contener su línea AS2

42 Caso De Uso Pedidos No Procesados

Caso de Uso:	Pedidos no procesados
Actores:	Operario
Objetivo:	Obtener los datos de los pedidos que no han sido consolidados en el almacén.
Precondiciones:	El pedido debe contener su línea XDI

De Recepciones

43 Caso De Uso Gestión De Caja

Caso de Uso:	Gestión de caja
Actores:	Operario
Objetivo:	Abrir, cerrar o crear una nueva caja para consolidar pedidos.
Precondiciones:	Si se abre o se cierra una nueva caja, ésta deberá haber sido creada.

44 Caso De Uso Gestión De Contenedor

Caso de Uso:	Gestión de contenedor
Actores:	Operario
Objetivo:	Abrir, cerrar o crear un nuevo contenedor para incluir cajas en él.
Precondiciones:	Si se abre o se cierra un nuevo contenedor, éste deberá haber sido creado.

45 Caso De Uso Recepción De La Mercancía

Caso de Uso:	Recepción de la mercancía
Actores:	Operario
Objetivo:	Crear la línea XDI para saber que la mercancía ha llegado al almacén.
Precondiciones:	El pedido debe contener la línea AS2.

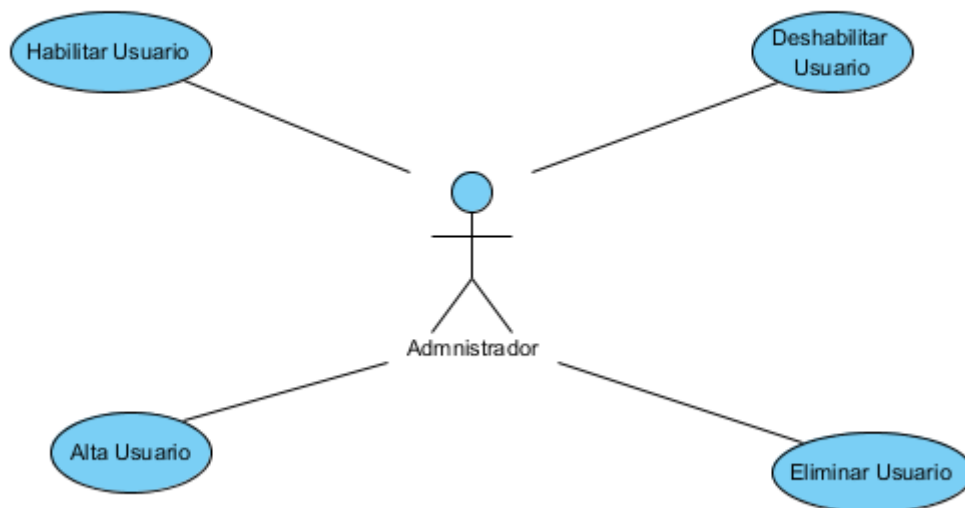
46 Caso De Uso Consolidación

Caso de Uso:	Consolidación
Actores:	Operario
Objetivo:	Emballar el pedido en su correspondiente caja y contenedor para procesar su envío posteriormente.
Precondiciones:	El pedido debe contener la línea XDI.

47 Caso De Uso Envío

Caso de Uso:	Envío
Actores:	Operario
Objetivo:	Seleccionar los contenedores se va a proceder en dicho envío.
Precondiciones:	La mercancía debe haber sido consolidada.
Post condiciones:	Es generada la línea XDS

6.2.1.1. *Administrador*



20 Diagrama De Casos De Uso Administrador

48 Caso De Uso Alta Usuario

Caso de Uso:	Alta Usuario
Actores:	Administrador
Objetivo:	Dar de Alta un nuevo usuario en el sistema.

49 Caso De Uso Habilitar Usuario

Caso de Uso:	Habilitar Usuario
Actores:	Administrador
Objetivo:	Habilitar un usuario en el sistema para que pueda conectarse a él.
Precondiciones:	El usuario debe existir y estar deshabilitado en el sistema

50 Caso De Uso Deshabilitar Usuario

Caso de Uso:	Deshabilitar Usuario
Actores:	Administrador
Objetivo:	Deshabilitar un usuario en el sistema para que no pueda conectarse a él.
Precondiciones:	El usuario debe existir y estar habilitado en el sistema

51 Caso De Uso Eliminar Usuario

Caso de Uso:	Eliminar Usuario
Actores:	Administrador
Objetivo:	Eliminar por completo un usuario del sistema.
Precondiciones:	El usuario debe existir en el sistema

7.2.4 Identificación de Clases Asociadas a un Caso de Uso

6.2.1.2. Operario

Consultas:

- Enviados. Relacionado con la clase Pedido. Para poder enlazar con el método ObtenerPedidosEnviados().
- Pedidos pendientes de recepción. Relacionado con la clase Pedido. Para poder enlazar con el método ObtenerPedidosPendientes().
- Pedidos no procesados. Relacionado con la clase Pedido. Para poder enlazar con el método ObtenerPedidosNoProcesados().

Gestión Incidencias: Relacionado con la clase pedido para poder llamar al método AltaIncidencia().

Recepciones:

- Gestión Caja. Relacionado con la clase Caja.
- Consolidación. Relacionado con las clases Caja, Pedido y Cargamento. Para poder realizar un envío incluyendo los cargamentos con sus cajas y pedidos.
- Gestión Contenedor. Relacionado con la clase Cargamento.
- Recepción Mercancía. Relacionado con la clase Pedido. Para poder crear la línea XDI.
- Envío. Relacionado con la clase Envío y Pedido. Para poder generar la línea XDS.

6.2.1.1. Administrador

El administrador se relaciona con la clase Usuario con los métodos de:

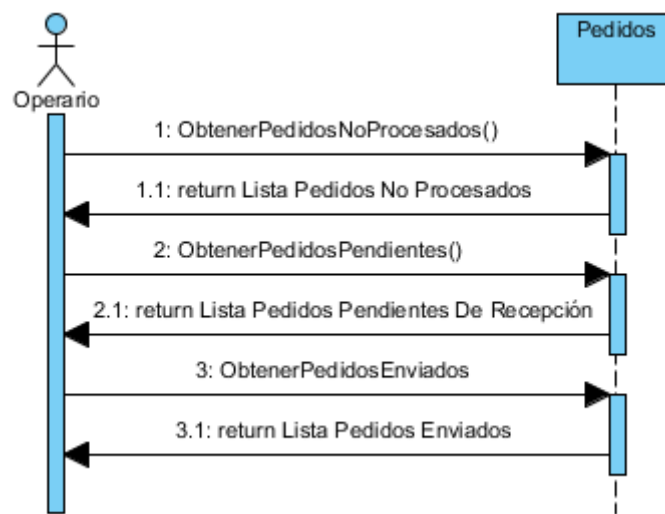
- AltaUsuario
- DeshabilitarUsuario

- HabilitarUsuario
- EliminarUsuario

7.2.5 Descripción de la Interacción de Objetos

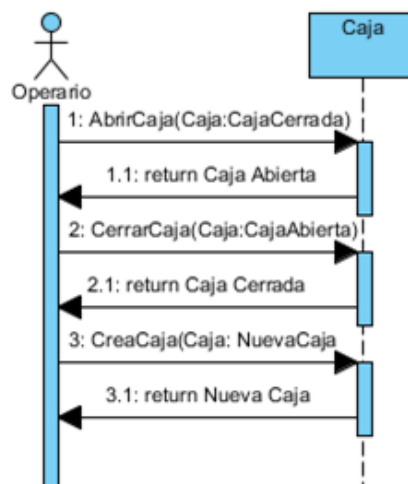
6.2.1.2. Operario

Consulta De Pedidos:



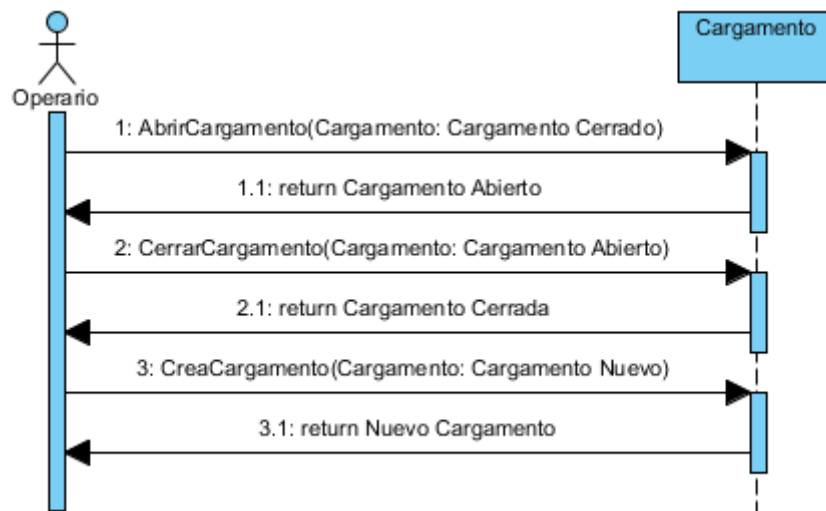
21 Diagrama De Secuencia Consulta De Pedidos

Gestión De Caja:



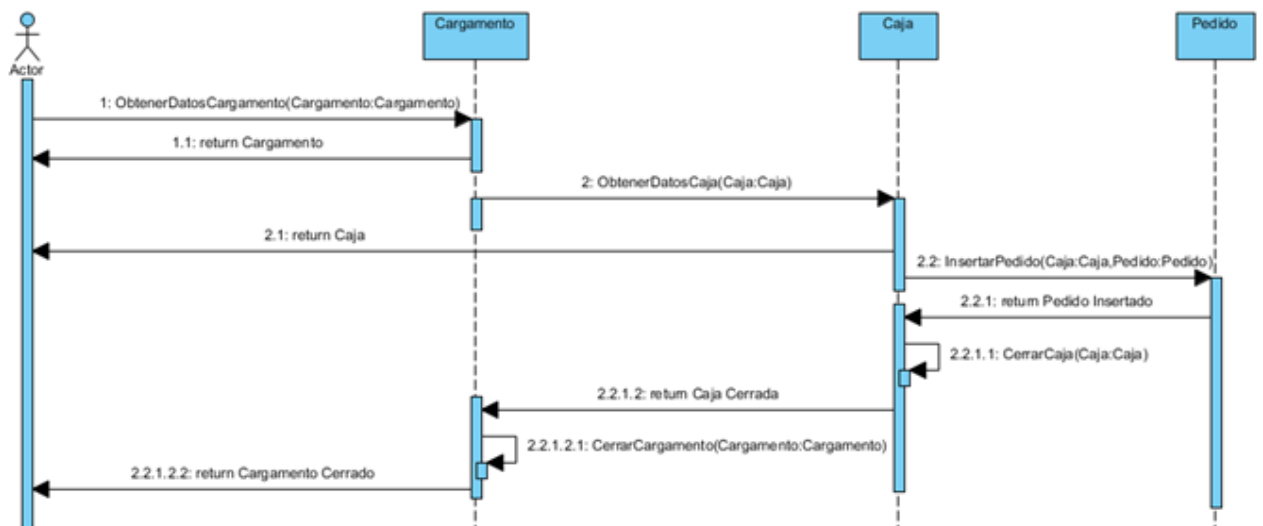
22 Diagrama De Secuencia Gestión De Cajas

Gestión De Cargamento:



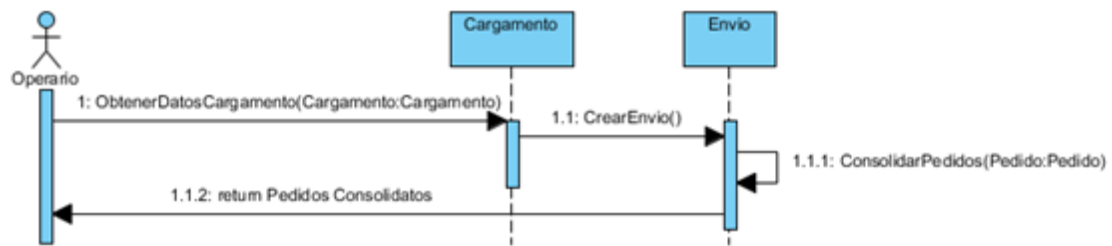
23 Diagrama De Secuencia Consulta De Cargamentos

Consolidación De Pedidos:



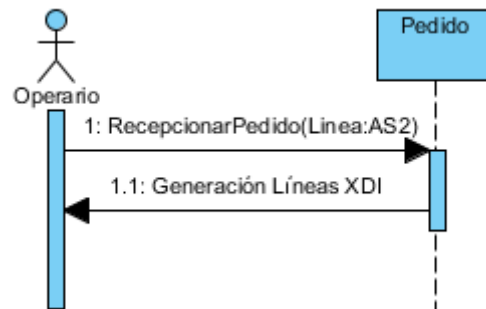
24 Diagrama De Secuencia Consolidación De Pedidos

Envíos:



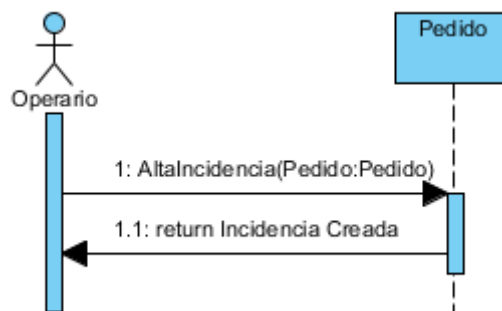
25 Diagrama De Secuencia Envíos

Recepción de pedidos:



26 Diagrama De Secuencia Recepción De Pedidos

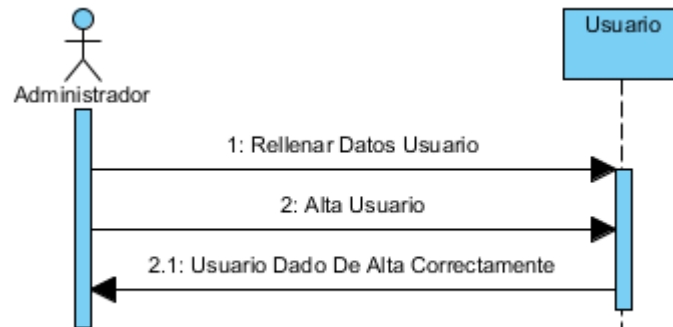
Incidencias:



27 Diagrama De Secuencia De Alta Incidencias

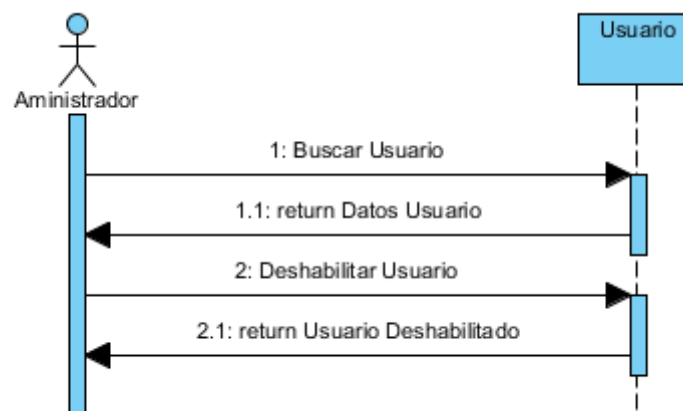
6.2.1.1. *Administrador*

Alta Usuario:



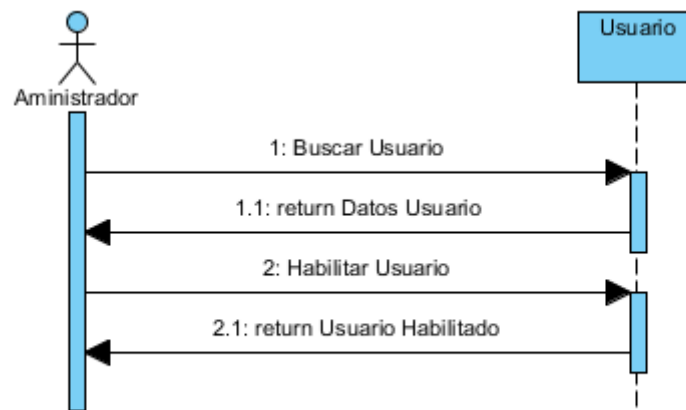
28 Diagrama De Secuencia De Alta Usuarios

Deshabilitar Usuario:



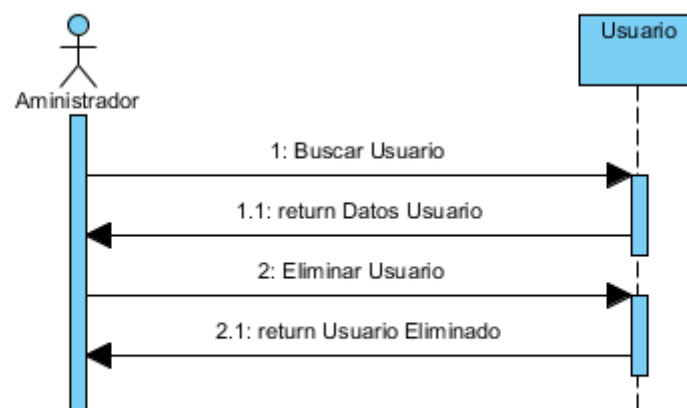
29 Diagrama De Secuencia Deshabilitar Usuarios

Habilitar Usuario:



30 Diagrama De Secuencia Habilitar Usuarios

Eliminar Usuario:

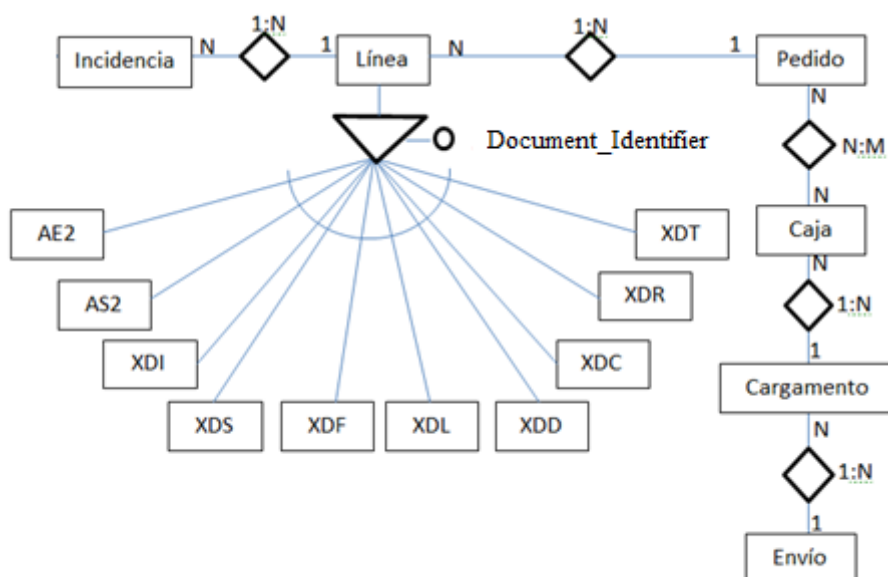


31 Diagrama De Secuencia Eliminar Usuarios

7.3 Elaboración del modelo de datos

7.3.1 Elaboración del Modelo Lógico de Datos

A continuación se muestra el modelo entidad-relación de la base de datos.



32 Modelo Entidad-Relación De La Base De Datos

En el modelo de datos la relación más importante es la jerarquía con exclusividad, en la cual, se recoge el núcleo de la información logística.

En la parte principal de la jerarquía se recoge la parte común de información de todas las líneas y en función del valor del identificador del documento se discrimina el tipo de línea que es y se almacena la información característica de cada una en su lugar correspondiente.

La relación entre las líneas y las incidencias es una relación 1:N debido a que una línea puede tener una o más incidencias y cada incidencia está asociada solamente a una línea.

La relación entre las incidencias y las soluciones es una relación N:M debido a que una incidencia puede tener una o más soluciones y cada solución puede aplicarse a varias soluciones.

La relación entre las líneas y los pedidos es una relación 1:N debido a que cada pedido tiene asociadas varias líneas y cada línea está asociada a un solo pedido.

La relación entre las cajas y los pedidos es una relación N:M debido a que un pedido puede estar dividido en varias cajas y cada caja puede contener varios pedidos.

La relación entre las cajas y los cargamentos es una relación 1:N debido a que cada cargamento puede llevar varias cajas y una caja está asociada a un solo cargamento.

La relación entre los cargamentos y los envíos es una relación 1:N debido a que cada envío puede llevar varios cargamentos y que un cargamento está asociado a un solo envío.

A continuación se muestra el modelo relacional de la base de datos:



7.3.2 Especificación de Necesidades de Migración de Datos y Carga Inicial

La carga inicial de datos se apoyará en la creación de la base de datos, cuyo código se encuentra establecido en el apartado *Diseño Físico de Datos* de dicho documento. En dicho script son incluidos los registros propios de las líneas FMS de la base de datos.

Es necesario tener el servidor de base de datos Sql Server 2008 para poder incluir el script de creación y ejecutarlo sin problemas.

De esta forma son generados los datos necesarios para arrancar la aplicación.

7.4 Estructura digital de los documentos electrónicos

Todas las solicitudes FMS deben estar preparadas en el formato del Standard MILSTRIP. A continuación se muestra una imagen del formulario de petición de material en el cual se especifican los valores concretos para dicha petición. Estos valores dan origen a la formación de la primera línea digital de pedido, **MILSTRIP-AE2**. En este apartado se especifican la localización exacta de cada elemento como posición del registro (rp) y sus posibles valores, en función del formato del documento digital al que pertenecen.

34 Identificación De Las Líneas FMS

BLOCK 3:
Enter the title, identification number, edition (year), and page number of the manufacturer's catalog in which the requested item is described.

BLOCK 6:
Enter the number (if known) that identifies the technical manual in which the requested item is described. (Include the applicable page number.)

BLOCK 4:
Enter the date of the manufacturer's catalog in calendar date format (year, month, day)

BLOCK 2:
Enter the name and address (including zip code, if known) of the manufacturer of the item when the CAGE code is not available.

BLOCK 1:
Enter the Commercial and Government Entity (CAGE) code for the item (when available). Also enter the complete part number when the part number exceeds 10 digits.

DOCUMENT NUMBER		ROUTING NUMBER		SP A N		ITEM IDENTIFICATION* (FOR FMS/Part No. Order)		UNIT OF ISSUE		QUANTITY		DOCUMENT NUMBER	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35		36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100		101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200		201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300							
A 0 5 N 6 5		9 8 5 3 4 N 3 1 2 2 9 8 8 4		E A 0 0 0 0 5 P L T 4 4									
DOCUMENT NO. (Item 1)		SUPPLEMENTARY ADDRESS		FUND CODE		DISTRIBUTION CODE		PROJECT CODE		PRIORITY		REQUIRED DELIVERY DAY OF YEAR	
8 1 8 6 2 2 4 2		N P A 5 J P A L		1 H		0 8		2 D					
DATE		SERIAL		FUND CODE		DISTRIBUTION CODE		PROJECT CODE		PRIORITY		REQUIRED DELIVERY DAY OF YEAR	
8 1 8 6 2 2 4 2		N P A 5 J P A L		1 H		0 8		2 D					
11. MANUFACTURER'S CODE AND PART NO. (When they exceed call columns 8 thru 23)		2. MANUFACTURER'S NAME		4. DATE (YYMMDD)		5. TECHNICAL ORDER NUMBER							
B & W PARTS CATALOG, P. 174A		BARCOCK & WILCOX CO.		96/01/22		N676412Y							
6. TECHNICAL MANUAL NUMBER		7. NAME OF ITEM REQUESTED											
NAVSEA TECHNICAL MANUAL B51-0048		ELEMENT, SOOT BLOWER, UNIT A											
8. DESCRIPTION OF ITEM REQUESTED		8a. COLOR		8b. SIZE		8c. SOURCE OF SUPPLY							
ELEMENT FOR USE WITH B&W SOOT BLOWER (B&W MODEL NUMBER SB2000, PART NO. SB20001042A)		N/A		N/A									
9. END ITEM APPLICATION		9a. MODEL NUMBER		9b. SERIAL NUMBER									
10. REQUISITIONER (Giver text name and address)		11. REMARKS											

DD FORM 1348-6, FEB 85 Edition of Apr 77 may be used until exhausted **DD SINGLE LINE ITEM REQUISITION SYSTEM DOCUMENT (MANUAL - LONG FORM)**

BLOCK 8:
Describe the item requested. Attach pictures or photographs when available. Attach additional pages if more space is need to fully describe the item.

BLOCK 8a:
Enter the color of the item requested, if applicable.

BLOCK 8b:
Enter the size of the item requested, if applicable.

BLOCK 7:
Enter the name of the item requested.

BLOCK 5:
Enter the number of the technical order or directive in which the requested item is described.

Código Identificador del Documento (RP 1-3)

El Código Identificador del Documento describe el propósito del documento. El código de la serie "A0_" (A "cero" _) indica un documento de solicitud MILSTRIP.

- A. A01 – Indica que la solicitud es para un elemento estándar (Número stock). También indica que no hay información en el área de observaciones.
- B. A04- Indica que la solicitud es para las publicaciones o formularios del ejército estadounidense. También es usado en los elementos de las solicitudes con el ejército estadounidense, Número de control del elemento, NICNs (Navy Item Control Numbers).
- C. A05- Indica que la solicitud puede ser para un elemento que no está en el estándar o para un elemento dentro del estándar con información incluida en el área de observaciones (la solicitud "A05" no puede ser enviada vía DDN o ILCS. La solicitud "A05" puede ser enviada vía STARR/PC.)

Para Canadá y México, el Identificador de solicitud del documento son: A0A es A01, A0D es A04 y A0E es A05.

Código identificador de ruta (rp 4-6)

Por ejemplo le código "N46" es el código para la Armada.

020204 Código de estado (rp 7)

Deja la posición 7 en blanco.

Número de stock /SMIC (rp 8-22)

En estas 13 posiciones se introduce el Número de stock nacional (NSN). Este número está asignado a cada elemento del estándar. Las posiciones 21 y 22 son usadas para la identificación de un material especial (SMIC) cuando es necesario. Cuando un NSN se muestra como el documento MILSTRIP aparece un guion entre las posiciones 4 y 5, 6 y 7, 9 y 10. Un NSN que aparece como: 5905-00-410-3897 puede ser introducido en un documento MILSTRIP como: 5905004103897.

El campo número de stock también puede ser usado para que los ejércitos introduzcan un número de control (NICN). Este no es el NSN, es un número utilizado para identificar el material principalmente por cada ejército. A continuación se muestran tres ejemplos, publicación, formulario y directiva. Para cada uno de los materiales, letras, sin números, son introducidos en las posiciones 5-6, (por ejemplo: "LP" para publicaciones, "LF" para formularios o "LD" para directivas).

Código unitario de expedición (rp 23-24)

Estas dos letras del código muestran la unidad de medida del material que ha sido solicitado. A continuación se muestran los códigos de expedición más comunes y su abreviatura en MILSTRIP.

Abreviatura	Descripción
EA	Unitario
PD	Caja
RL	Carrete
FT	Pies

Cantidad (RP 25-29)

La cantidad será introducida en las posiciones 25-29. Se mostrarán el número de cantidades de material que han sido solicitadas. El cliente debe siempre considerar el campo cantidad y la unidad de medida. Por ejemplo, supongamos que el campo muestra 12. Si la unidad de medida es DZ (docena) se enviarán 12 docenas de material es decir 144 unidades. Para recibir 12 unidades con el valor del campo medida "DZ" el cliente deberá introducir "00001".

Número de documento (RP 30-43)

El número de documento ocupa las posiciones 30-43. Este número debe ser único para cada solicitud enviada. El valor de campo nunca debe ser usado más de una vez. El número de documento está dividido en cuatro secciones:

- A. Código de servicio (RP 30) – Introduce el código de servicio "P" en la posición 30 si se trata de una petición del para la armada estadounidense.
- B. Peticionario (RP 31-35) – En estos cinco caracteres se introducen los siguientes cuatro partes:
 - i. *Código de país (RP 31-32)* – Identifica el país por el cual el material está siendo solicitado.
 - ii. *Código para la huella (RP 33)* – Este código identifica donde el material tiene que ser entregado junto con el país del cliente FMS. Este código está designado por la LOA sino aparecerá un cero en su lugar.

- iii. *Código del término de entrega (RP 34)* - Este código indica los términos bajo los cuales los materiales serán entregados en el país del cliente.
- iv. *Tipo de asociación (RP 35)* – Este código identifica los acuerdos de pago que a los que se han llegado entre el cliente y los estados unidos que se aplicarán al cargamento FMS.
- C. *Fecha (RP 36 – 39)* – Es la fecha en formato juliano en la cual la petición es preparada. La posición 36 corresponde al último dígito del año.
- D. *Número de Serie (RP 40-43)* – El número de serie debe ser único con una fecha juliana y país.

Código de demanda (RP 44)

Se introduce el valor “R” si el código del tipo de asistencia en la columna 5 de la página 2 de la LOA es una “V”. En otro caso es “N”.

Código de dirección adicional (RP 45-50)

- A. *Código de Servicio (RP 45)* – Este código ejerce el cliente al que es enviada la petición. “P” si es la armada, “B” si es el ejército de tierra, “D” si es el ejército del aire.
- B. *Código de opción FMS Ofrecimiento/Liberación* – Este código debe ser el código de la columna 6 de la página 2 de la LOA. Indica como será servido el material al cliente o su gente y si será requerida una Notificación de Disponibilidad, “Notice of Availability” (NOA), antes de que el material tenga que ser servido.
- C. *Código de intermediario* – Este código se muestra en la página 1 de la LOA. Identifica el primer destino del cargamento. El material no siempre está entregado al intermediario.
- D. *Indicador de Cargamento FMS (RP 48-50)* – Es el término que se utiliza para identificar el cargamento FMS. Por ejemplo, este se el que se muestra en las últimas 3 letras de la LOA, LT-P-**ABC**.

Código de señal (RP 51)

El valor “L” indica que el material será enviado a la dirección indicada en la dirección adicional.

Código de fondo (RP 52 -53)

Ayuda a asociar el registro a la petición.

Código de distribución (RP 54 -56)

La posición 54 va en blanco, para los materiales que tienen NSN, el valor será un valor del determinado por el COG, para materiales que no tengan NSN las posiciones 55-56 tendrán uno de los siguientes valores:

Material	Valor
Material naval	1H
Material electrónico	1H
Material aeronáutico	1R
Dispositivos de entrenamiento	2O
Publicaciones	OI
Otros	1R

Código de proyecto (RP 57-59)

Un cliente FMS puede usar el código de proyecto sobre una petición para un proyecto o programa especial. Debe ser obtenido por el ejército que envía el material. Puede ir en blanco.

Prioridad (RP 60-61)

Utiliza las reglas de UMMIPS para obtener la prioridad.

Fecha de requerimiento de entrega (RP 62-64)

Formato de fecha juliana.

Código de aviso (RP 65-66)

Se utiliza para indicar que hay que realizar alguna operación especial sobre el material.

Posiciones de libre disposición (Vacías) (RP 67- 73)

Precio unitario (RP 74-80)

El precio siempre viene dado en dólares americanos.

Guía adicional para peticiones de material no estándar

Los datos del MILSTRIP para las peticiones de un material no estándar son muy similares a los del material estándar, salvo alguna modificación:

- El código identificador de documento es “A05” en las posiciones 1-3
- Se introduce el código de la entidad gubernamental y comercial “Commercial and Government Entity”(CAGE) del fabricante en las posiciones 8-12.
- Se introduce el número identificador del elemento en las posiciones 13-22, si este número ocupa más de diez posiciones se utilizará el apartado de observaciones.

Tipos de documentos de estado

El número de documento (rp 30-43) siempre será exactamente el mismo que va en la petición original y el código de identificación de documento (rp 1-3) es el que se usa para distinguir el documento de petición de un documento de estado. Existen cuatro estados:

- A. Código Identificador de documento “AE2” – Será el primer documento de estado enviado para las peticiones “**pull**”. Con él se notificará al cliente de que el ejército ha recibido la petición del cliente.
- B. Código Identificador de Documento “AE-” – Es para las peticiones “**push**” e indica al cliente que el ejército al que se le realizó la petición ha recibido y procesado la petición desde el sistema de suministro de Estados Unidos.
- C. Código identificador de Documento “AS2” – Cuando el sistema de suministro envía el material se envía este documento.

Como interpretar el Estado del Suministro

El propósito del documento de estado de suministro es indicar mediante el código de estado introducido en las posiciones 65-66 el estado en que se encuentra la petición realizada. Sin embargo hay varios campos que también tienen información crítica:

Posiciones	Información proporcionada
8-22	Número Stock Nacional (NSN)
23-24	Unidades de material
25-29	Cantidad
44	Sufijo
62-64	Fecha en que se creo el documento

65-66	Acción a cometer
67-69	Actividad de envío
70-73	Fecha en la que será enviado el material
74-80	Precio unitario

Como interpretar el estado de envío

Código identificador del documento “AS2” indica que el material ha sido enviado al cliente.

Posición	Información Proporcionada
4-6	Código identificador de ruta
25-29	Cantidad
44	Sufijo
57-59	Fecha en la que el material ha sido enviado
62-76	Campo identificador de envío
77	Este código indica como fue enviado el material

Código de sufijo

A veces el sistema de suministro debe hacer dos o más envíos para proporcionar el total de la cantidad el elemento solicitado. El código de sufijo es usado para mostrar que el envío que está siendo enviado en varias partes.

Campo de Identificador de cargamento

Este campo muestra el Número de Control de Transporte (TCN):

- Número de cargamento para la cuenta del gobierno (B)
- Número de correo certificado (C)
- Número del seguro (I)
- Número de registro del correo (R)

Proceso de seguimiento “Follo-Up”

Se trata de un documento de estado que indica que hay información que pasado el plazo no ha sido enviada y requiere que vuelva a ser enviada.

- Follow-Up AF1 – Actualización de estado

- Follow-Up AT1, AT4, AT5 – Reenvío de información
- Follow-Mejora de fecha estimada de envío

Incremento de cantidad

Hay que realizar una petición nueva para solicitar más cantidad de un material.

Cancelación de peticiones

Si el material ya ha sido enviado cuando el ejército recibe la cancelación “AC1” entonces envía un “AU2” para indicar que no es posible la cancelación.

Documento de modificación de petición

Con los códigos “AM1”, “AM4” o “AM5” se pueden realizar ciertos cambios de las peticiones:

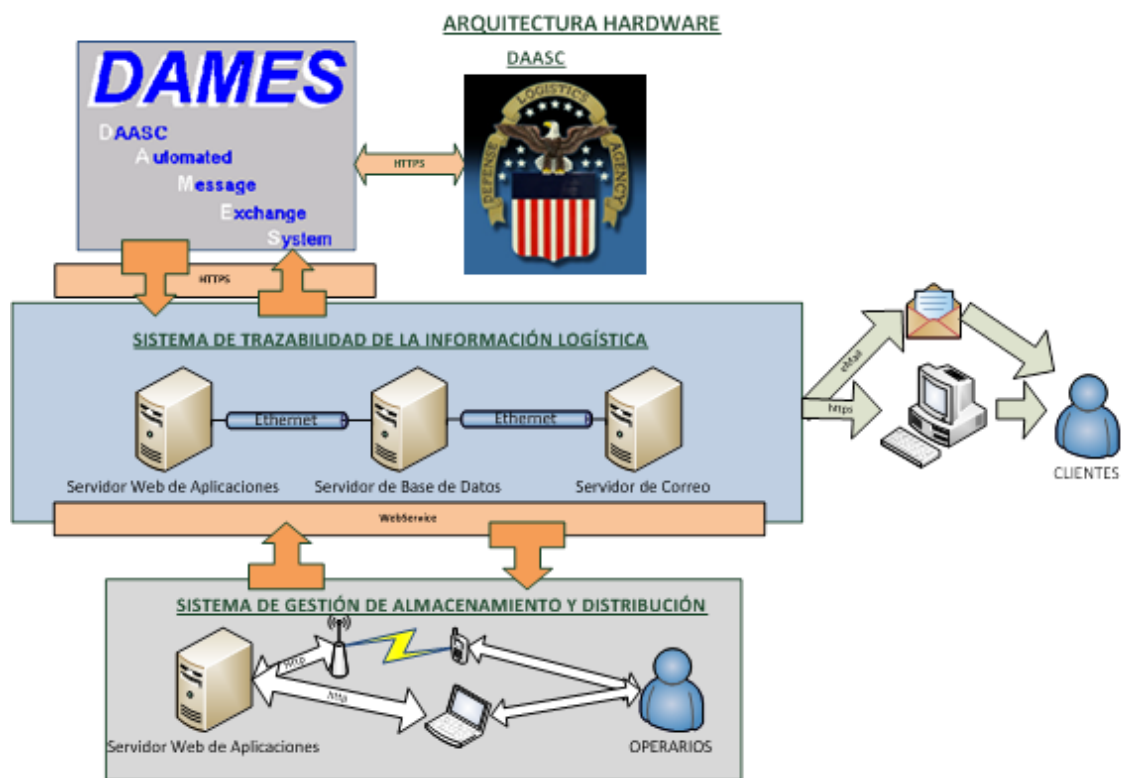
- FMS Código de oferta (rp 46)
- Código de intermediario (rp 47)
- Código de proyecto (rp 57-59)
- Prioridad (rp 60-61)
- Fecha de entrega (rp 62-64)
- Código de aviso (rp 65-66)

8 Diseño del sistema

En este apartado se definirá la especificación de la arquitectura y de la base de datos de una manera precisa para que la programación verifique el catálogo de requisitos y la especificación funcional producida en el análisis.

8.1 Definición de la arquitectura del sistema

8.1.1 Definición de Niveles de Arquitectura



35 Niveles De La Arquitectura Del Sistema

Tal como refleja la imagen, tenemos tres niveles a describir.

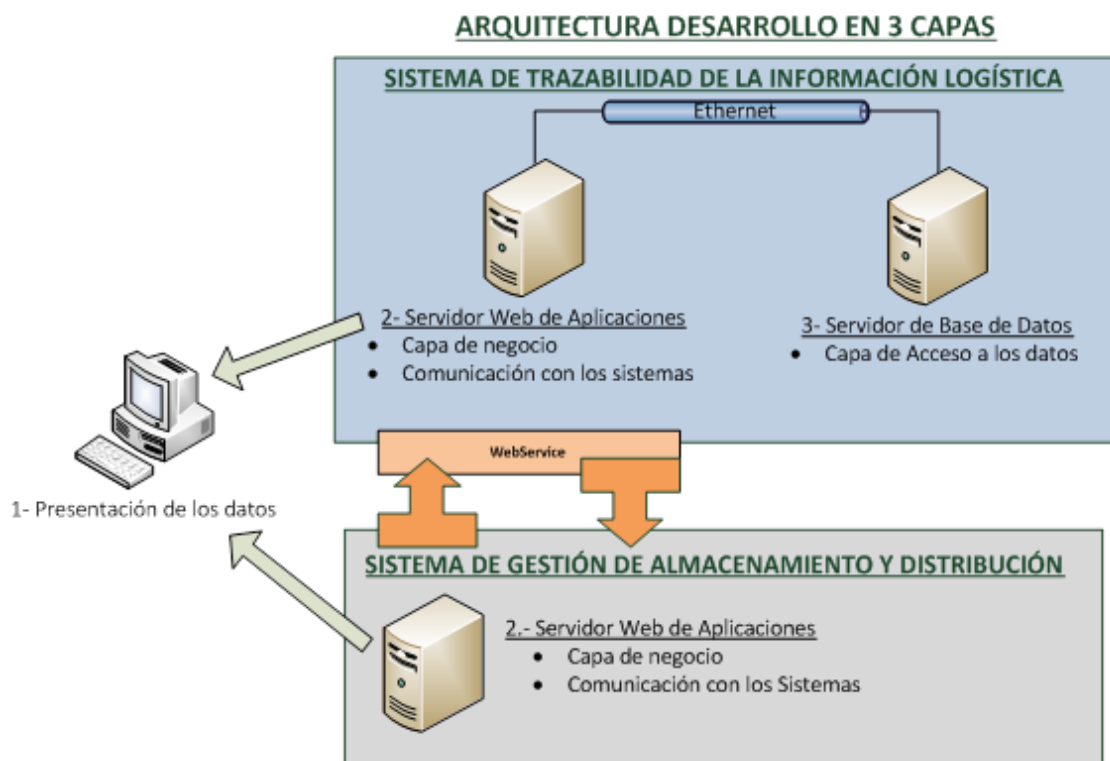
El nivel superior corresponde a la interfaz Dames la cual es encargada de la actualización de las líneas correspondientes a los pedidos solicitados por los clientes.

Éste nivel enlaza con el segundo en el cual son guardadas dichas líneas en la base de datos avisando a los clientes que el proceso de envío de las mercancías ha comenzado mediante correo electrónico. Dicha comunicación se realiza vía protocolo HTTPS

Dichas líneas generadas valdrán a la aplicación de almacén para poder consolidar dichas mercancías, actualizando las nuevas líneas en el servidor de la base de datos. Dicha comunicación se realiza vía Servicios Web

Así todo el proceso está completamente enlazado de tal forma que el cliente puede seguir una trazabilidad del curso de sus pedidos y mercancías.

8.1.2 Especificación del Entorno Tecnológico



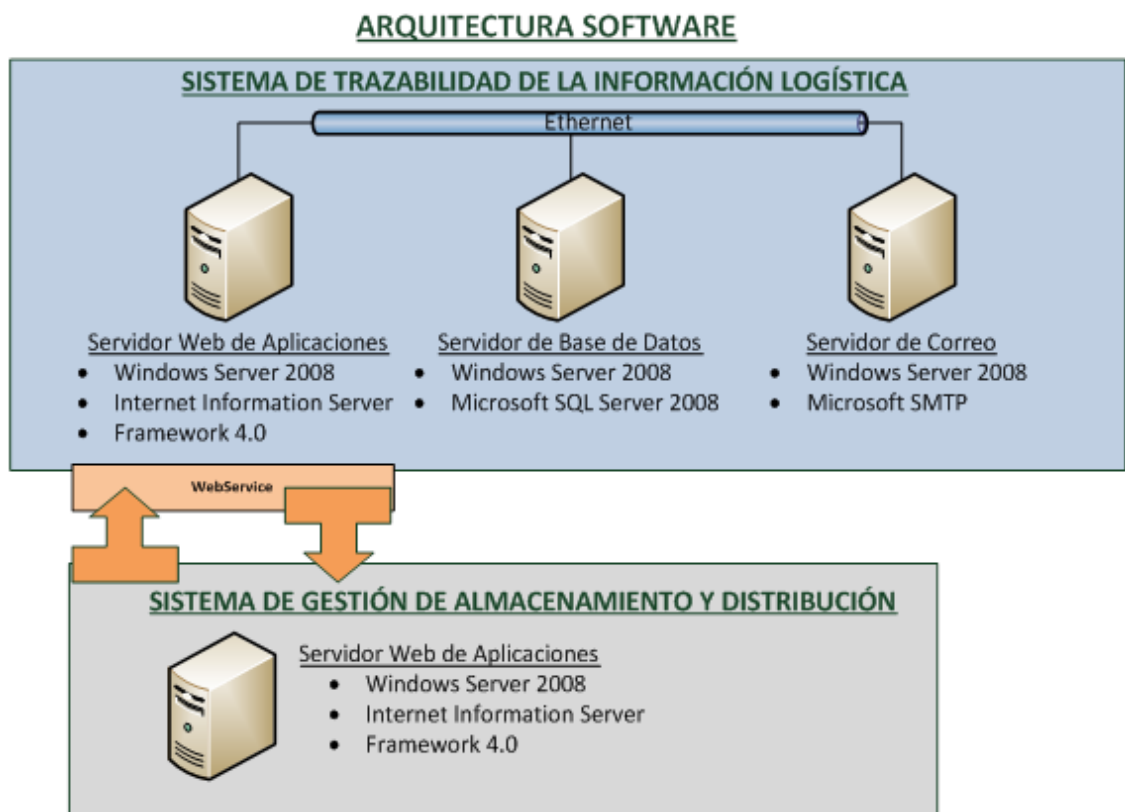
36 Entorno Tecnológico Desarrollo En 3 Capas

El entorno tecnológico se divide en tres capas también.

1. Presentación de los datos: son las interfaces de la aplicación que van a suministrar al cliente toda la información necesaria en cuanto a sus pedidos. Dicha capa se nutre del servidor web de aplicaciones para cargar el software el cual a través suyo enlaza

con el servidor de la base de datos para obtener los datos solicitados.

2. Sistema de gestión de almacenamiento y distribución la cual se compone de la capa de negocio y la comunicación de la aplicación a usar.
3. Sistema de trazabilidad de la información logística en la cual se concentra el servidor de aplicaciones para poder acceder al servidor de la base de datos. La comunicación entre los servidores se realiza vía servicios web.



37 Arquitectura Software Del Sistema

En este cuadro se puede contemplar los requerimientos Software que se incluyen en dicho proyecto junto con un proceso de trazabilidad implementado.

El servidor web de aplicaciones contiene Windows Server 2008 con la instalación del servidor Internet InformationServer y el Framework 4.0.

El servidor de la base de datos también incluye Windows Server 2008 y el gestor de base de datos Microsoft Sql Server 2008.

El servidor de correo también incluye el sistema operativo Windows Server 2008 y el protocolo de comunicación Microsoft SMTP para la recepción y envío de correos.

La comunicación entre el servidor web de aplicaciones y el servidor de la base de datos se realiza vía servicios web.

En la aplicación diseñada no se realizará ni implementarán los servicios web ya que no existe comunicación con el sistema de trazabilidad, ya que dicho sistema pertenece a otro módulo distinto.

8.2 Requisitos Del Sistema

A continuación se muestran los requisitos de sistema ó software obtenidos:

8.2.1 Funcionales

52 Requisito Sistema FSR -01

ID: FSR –01	Tipo: Sistema	Necesidad: Alta
Título: Administración De Usuarios		
Fuente: Cliente		
Descripción: El sistema guardará los cambios que se introduzcan en los usuarios por parte del administrador en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC – 0001, UC – 0019		
Relación Con Requisitos De Sistema: FSR-02		

53 Requisito Sistema FSR -02

ID: FSR -02	Tipo: Sistema	Necesidad: Alta
Título: Alta De Usuarios		
Fuente: Cliente		
Descripción: El sistema guardará los nuevos usuarios que introduzca el administrador comprobando que sus datos son correctos		
Estabilidad: Alta		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UC – 0001, UC – 0019		
Relación Con Requisitos De Sistema: FSR-01		

54 Requisito Sistema FSR -03

ID: FSR -03	Tipo: Sistema	Necesidad: Alta
Título: Codificación Contraseñas		
Fuente: Desarrollador		
Descripción: El sistema guardará las contraseñas de los usuarios codificadas en la base de datos.		
Estabilidad: Media		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UR – 0024		
Relación Con Requisitos De Sistema: FSR-01		

55 Requisito Sistema FSR -04

ID: FSR -04	Tipo: Sistema	Necesidad: Alta
Título: Caducidad De Contraseñas		
Fuente: Cliente		
Descripción: Las contraseñas caducarán cada cierto tiempo avisando al usuario para que pueda modificarla.		
Estabilidad: Media		
Verificabilidad: Medio		

Relación Con Requisitos De Usuario: UR – 0023
--

Relación Con Requisitos De Sistema: FSR-03

56 Requisito Sistema FSR -05

ID: FSR -05	Tipo: Sistema	Necesidad: Alta
Título: Alta De Cargamentos		
Fuente: Cliente		
Descripción: Se comprobarán que los datos del cargamento son correctos y se insertará en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UC-0002		
Relación Con Requisitos De Sistema: FSR-06		

57 Requisito Sistema FSR -06

ID: FSR -06	Tipo: Sistema	Necesidad: Alta
Título: Clave Única En Cargamentos		
Fuente: Cliente		
Descripción: Los cargamentos se generarán mediante una clave única e irreplicable, para su posterior guardado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Media		
Relación Con Requisitos De Usuario: UC-0002		
Relación Con Requisitos De Sistema: FSR-05		

58 Requisito Sistema FSR -07

ID: FSR -07	Tipo: Sistema	Necesidad: Alta
Título: Cierre De Cargamentos		
Fuente: Cliente		
Descripción: Al cerrar un cargamento existente, se cambiará de estado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-003		
Relación Con Requisitos De Sistema: FSR -08		

59 Requisito Sistema FSR -08

ID: FSR -08	Tipo: Sistema	Necesidad: Alta
Título: Cierre De Cargamentos-Caja		
Fuente: Cliente		
Descripción: Al cerrar un cargamento existente, se comprobará que todas las cajas insertadas en él estén cerradas.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-003		
Relación Con Requisitos De Sistema: FSR -07		

60 Requisito Sistema FSR -09

ID: FSR -09	Tipo: Sistema	Necesidad: Alta
Título: Abrir Cargamento		
Fuente: Cliente		
Descripción: Al abrir un cargamento existente, se cambiará de estado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-004		
Relación Con Requisitos De Sistema: FSR -07		

61 Requisito Sistema FSR -10

ID: FSR -10	Tipo: Sistema	Necesidad: Alta
Título: Alta De Caja		
Fuente: Cliente		
Descripción: Se comprobarán que los datos de la caja son correctos y se insertará en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UC-005		
Relación Con Requisitos De Sistema: FSR-11		

62 Requisito Sistema FSR -11

ID: FSR -11	Tipo: Sistema	Necesidad: Alta
Título: Clave Única En Caja		
Fuente: Cliente		
Descripción: Las cajas se generarán mediante una clave única e irrepitable, para su posterior guardado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Media		
Relación Con Requisitos De Usuario: UC-005		
Relación Con Requisitos De Sistema: FSR-10		

63 Requisito Sistema FSR -12

ID: FSR -12	Tipo: Sistema	Necesidad: Alta
Título: Cierre De Caja		
Fuente: Cliente		
Descripción: Al cerrar una cana existente, se cambiará de estado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		

Relación Con Requisitos De Usuario: UC-0007

Relación Con Requisitos De Sistema: FSR -13

64 Requisito Sistema FSR -13

ID: FSR -13	Tipo: Sistema	Necesidad: Alta
Título: Cierre De Caja-Pedido		
Fuente: Cliente		
Descripción: Al cerrar una caja existente, se comprobará que tiene algún pedido insertado en ella.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UR-0026		
Relación Con Requisitos De Sistema: FSR -12		

65 Requisito Sistema FSR -14

ID: FSR -14	Tipo: Sistema	Necesidad: Alta
Título: Abrir Caja		
Fuente: Cliente		
Descripción: Al abrir una caja existente, se cambiará de estado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0006		
Relación Con Requisitos De Sistema: FSR -12		

66 Requisito Sistema FSR -15

ID: FSR -15	Tipo: Sistema	Necesidad: Alta
Título: Insertar Pedido En Caja		
Fuente: Cliente		
Descripción: El sistema guardará la información de la localización del pedido en dicha caja en		

la base de datos.
Estabilidad: Alta
Verificabilidad: Alta
Relación Con Requisitos De Usuario: UC-0008
Relación Con Requisitos De Sistema: FSR -12, FSR-16, FSR-17

67 Requisito Sistema FSR -16

ID: FSR -16	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Pedido Antes De Su Consolidación		
Fuente: Cliente		
Descripción: El sistema comprobará los datos del pedido que sean correctos para su inserción en caja. El peso del pedido y sus dimensiones deberán ser menos a la de la caja.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0008		
Relación Con Requisitos De Sistema: FSR -15, FSR-12,FSR-17		

68 Requisito Sistema FSR -17

ID: FSR -17	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Caja Antes De La Consolidación Del Pedido		
Fuente: Cliente		
Descripción: La caja deberá estar abierta para la inserción del pedido en ella.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0008		
Relación Con Requisitos De Sistema: FSR -16, FSR -15, FSR -14,		

69 Requisito Sistema FSR -18

ID: FSR -18	Tipo: Sistema	Necesidad: Alta
Título: Desasignar Pedido De Caja		
Fuente: Cliente		
Descripción: Al desasignar el pedido en caja, éste volverá a visualizarse en la aplicación para su posterior consolidación.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0009		
Relación Con Requisitos De Sistema: FSR -19		

70 Requisito Sistema FSR -19

ID: FSR -19	Tipo: Sistema	Necesidad: Alta
Título: Propiedad para Desasignar Pedido De Caja		
Fuente: Cliente		
Descripción: Para Desasignar un pedido incluido en una caja, ésta deberá abrirse antes de la operación.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0009		
Relación Con Requisitos De Sistema: FSR -18, FSR-14		

71 Requisito Sistema FSR -20

ID: FSR -20	Tipo: Sistema	Necesidad: Alta
Título: Insertar Caja En Cargamento		
Fuente: Cliente		
Descripción: El sistema guardará la información de la localización de la localización de la caja en dicho cargamento.		
Estabilidad: Alta		
Verificabilidad: Alta		

Relación Con Requisitos De Usuario: UC-0010
Relación Con Requisitos De Sistema: FSR -21

72 Requisito Sistema FSR -21

ID: FSR -21	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Caja Antes De Añadir A Un Cargamento		
Fuente: Cliente		
Descripción: El sistema comprobará que la caja a insertar en el cargamento está completamente cerrada.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0010		
Relación Con Requisitos De Sistema: FSR -20		

73 Requisito Sistema FSR -22

ID: FSR -22	Tipo: Sistema	Necesidad: Alta
Título: Desasignar Caja De Cargamento		
Fuente: Cliente		
Descripción: Al desasignar una caja insertada en un cargamento, ésta volverá a visualizarse en la aplicación para su posterior inserción en otro cargamento.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0011		
Relación Con Requisitos De Sistema: FSR -23		

74 Requisito Sistema FSR -23

ID: FSR -23	Tipo: Sistema	Necesidad: Alta
Título: Propiedad para Desasignar Caja De Cargamento		
Fuente: Cliente		

Descripción: Para Desasignar una caja incluido en un cargamento, éste deberá abrirse antes de la operación.
Estabilidad: Alta
Verificabilidad: Alta
Relación Con Requisitos De Usuario: UC-0011
Relación Con Requisitos De Sistema: FSR -22

75 Requisito Sistema FSR -24

ID: FSR -24	Tipo: Sistema	Necesidad: Alta
Título: Asignar Cargamento A Envío		
Fuente: Cliente		
Descripción: El sistema guardará la información de los cargamentos que va conteniendo dicho envío.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0012		
Relación Con Requisitos De Sistema: FSR -25		

76 Requisito Sistema FSR -25

ID: FSR -25	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Cargamento Antes De Añadirlo A Un Envío		
Fuente: Cliente		
Descripción: El sistema comprobará que el cargamento está cerrado para añadirlo al envío		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0012		
Relación Con Requisitos De Sistema: FSR -24		

77 Requisito Sistema FSR -26

ID: FSR -26	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Contenido Cargamento Antes De Añadirlo A Un Envío		
Fuente: Cliente		
Descripción: El sistema comprobará que el cargamento contiene al menos una caja.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0012, UR-0028		
Relación Con Requisitos De Sistema: FSR -24		

78 Requisito Sistema FSR -27

ID: FSR -27	Tipo: Sistema	Necesidad: Alta
Título: Desasignar Cargamento De Envío		
Fuente: Cliente		
Descripción: Al desasignar un cargamento insertado en un envío, éste volverá a visualizarse en la aplicación para su posterior inserción en otro envío.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0013		
Relación Con Requisitos De Sistema: FSR -28		

79 Requisito Sistema FSR -28

ID: FSR -28	Tipo: Sistema	Necesidad: Alta
Título: Propiedad para Desasignar Cargamento De Envío		
Fuente: Cliente		
Descripción: Para Desasignar un cargamento incluido en un envío, éste deberá abrirse antes de la operación.		
Estabilidad: Alta		
Verificabilidad: Alta		

Relación Con Requisitos De Usuario: UC-0013
Relación Con Requisitos De Sistema: FSR -27

80 Requisito Sistema FSR -29

ID: FSR -29	Tipo: Sistema	Necesidad: Muy Alta
Título: Recepción De Envíos		
Fuente: Cliente		
Descripción: Se insertará en la base de datos del almacén los pedidos que vayan llegando con sus propiedades usando la línea AS2 ó XDC ó XDS		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0014		
Relación Con Requisitos De Sistema: FSR -30, FSR-31		

81 Requisito Sistema FSR -30

ID: FSR -30	Tipo: Sistema	Necesidad: Muy Alta
Título: Recepción De Envío: Clave		
Fuente: Cliente		
Descripción: La clave de los pedidos ha insertar y utilizar en todo el proceso será el Código de la identificación de la línea con formato FMS. (Requisition Document Number).		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0014, UR-0021		
Relación Con Requisitos De Sistema: FSR -29, FSR-30		

82 Requisito Sistema FSR -31

ID: FSR -31	Tipo: Sistema	Necesidad :Muy Alta
Título: Generación De Línea En La Recepción		
Fuente: Cliente		
Descripción: Una vez recepcionada la mercancía se generará su línea correspondiente. XDI, XDL ó XDR		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0014, UR-0021		
Relación Con Requisitos De Sistema: FSR -29, FSR-30		

83 Requisito Sistema FSR -32

ID: FSR -32	Tipo: Sistema	Necesidad: Alta
Título: Consolidar Envío		
Fuente: Cliente		
Descripción: Al cerrar un envío, éste cambiará de estado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0015		
Relación Con Requisitos De Sistema: FSR-33, FSR-34, FSR-35		

84 Requisito Sistema FSR -33

ID: FSR -33	Tipo: Sistema	Necesidad: Alta
Título: Comprobar Consolidación Envío		
Fuente: Cliente		
Descripción: Al cerrar un envío, éste deberá contener mínimo un cargamento cerrado.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0015, UR-0027		

Relación Con Requisitos De Sistema: FSR-32, FSR-34, FSR-35

85 Requisito Sistema FSR -34

ID: FSR -34	Tipo: Sistema	Necesidad: Muy Alta
Título: Generación De Líneas De Envío		
Fuente: Cliente		
Descripción: Al cerrar un envío, se generará su línea correspondiente: XDS, XDD, ó XDT		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0015, UR-0021		
Relación Con Requisitos De Sistema: FSR-33, FSR-32, FSR-35		

86 Requisito Sistema FSR -35

ID: FSR -35	Tipo: Sistema	Necesidad: Muy Alta
Título: Identificación Del Envío.		
Fuente: Cliente		
Descripción: El envío será identificado mediante una clave única y no repetible.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0015		
Relación Con Requisitos De Sistema: FSR-34, FSR-33, FSR-32		

87 Requisito Sistema FSR -36

ID: FSR -36	Tipo: Sistema	Necesidad: Muy Alta
Título: Consulta De Pedidos No Procesados		
Fuente: Cliente		
Descripción: Mediante una búsqueda por la identificación del pedido (Requisition Document Number). Se obtendrán los pedidos que todavía no han sido procesados en el almacén, todavía no se han generado sus líneas correspondientes de salida: XDS, XDD ó XDT.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0016, UR-0021		
Relación Con Requisitos De Sistema: FSR-37		

88 Requisito Sistema FSR -37

ID: FSR -37	Tipo: Sistema	Necesidad: Muy Alta
Título: Existencia De Pedidos A Consultar		
Fuente: Cliente		
Descripción: Para obtener los datos de los pedidos que no han sido procesados o consolidados, éstos deberán haber sido antes recepcionados en el almacén, es decir que contendrán su línea correspondiente: XDI, XDL ó XDR.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0016, UR-0021		
Relación Con Requisitos De Sistema: FSR-36		

89 Requisito Sistema FSR -38

ID: FSR -38	Tipo: Sistema	Necesidad: Muy Alta
Título: Consulta De Pedidos No Recepcionados		
Fuente: Cliente		
Descripción: Mediante una búsqueda por la identificación del pedido (Requisition Document Number). Se obtendrán los pedidos que todavía no han sido recepcionados en el almacén, todavía no se han generado sus líneas correspondientes de salida: XDI, XDL ó XDR.		

Estabilidad: Alta
Verificabilidad: Alta
Relación Con Requisitos De Usuario: UC-0017, UR-0021
Relación Con Requisitos De Sistema: FSR-39

90 Requisito Sistema FSR -39

ID: FSR -39	Tipo: Sistema	Necesidad: Muy Alta
Título: Existencia De Pedidos A Consultar		
Fuente: Cliente		
Descripción: Para obtener los datos de los pedidos que no han sido recepcionados, éstos deberán contener su línea correspondiente en la base de datos, es decir que contendrán su línea: AS2, XDS, XDC ó XDT		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0017, UR-0021		
Relación Con Requisitos De Sistema: FSR-38		

91 Requisito Sistema FSR -40

ID: FSR -40	Tipo: Sistema	Necesidad: Muy Alta
Título: Crear Envío		
Fuente: Cliente		
Descripción: Se comprobarán que los datos del envío son correctos y se insertará en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Alta		
Relación Con Requisitos De Usuario: UC-0018		
Relación Con Requisitos De Sistema: FSR -41		

92 Requisito Sistema FSR -41

ID: FSR -41	Tipo: Sistema	Necesidad: Alta
Título: Clave Única En Envío		
Fuente: Cliente		
Descripción: Los envíos se generarán mediante una clave única e irrepetible, para su posterior guardado en la base de datos.		
Estabilidad: Alta		
Verificabilidad: Media		
Relación Con Requisitos De Usuario: UC-0018		
Relación Con Requisitos De Sistema: FSR-40		

93 Requisito Sistema FSR -42

ID: FSR -42	Tipo: Sistema	Necesidad: Alta
Título: Alta Incidencia		
Fuente: Cliente		
Descripción: Se rellenarán los campos de la incidencia junto con su tipo y se dará de alta la incidencia asociada al pedido seleccionado.		
Estabilidad: Alta		
Verificabilidad: Media		
Relación Con Requisitos De Usuario: UC-0020		

8.2.2 No Funcionales

94 Requisito Sistema NFSR -43

ID: NFSR -43	Tipo: Sistema	Necesidad: Alta
Título: Tecnología		
Fuente: Cliente		
Descripción: La aplicación soportará los navegadores más comunes y utilizados: Internet Explorer, Mozilla Firefox		
Estabilidad: Alta		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UR-0022		
Relación Con Requisitos De Sistema: -		

95 Requisito Sistema NFSR -44

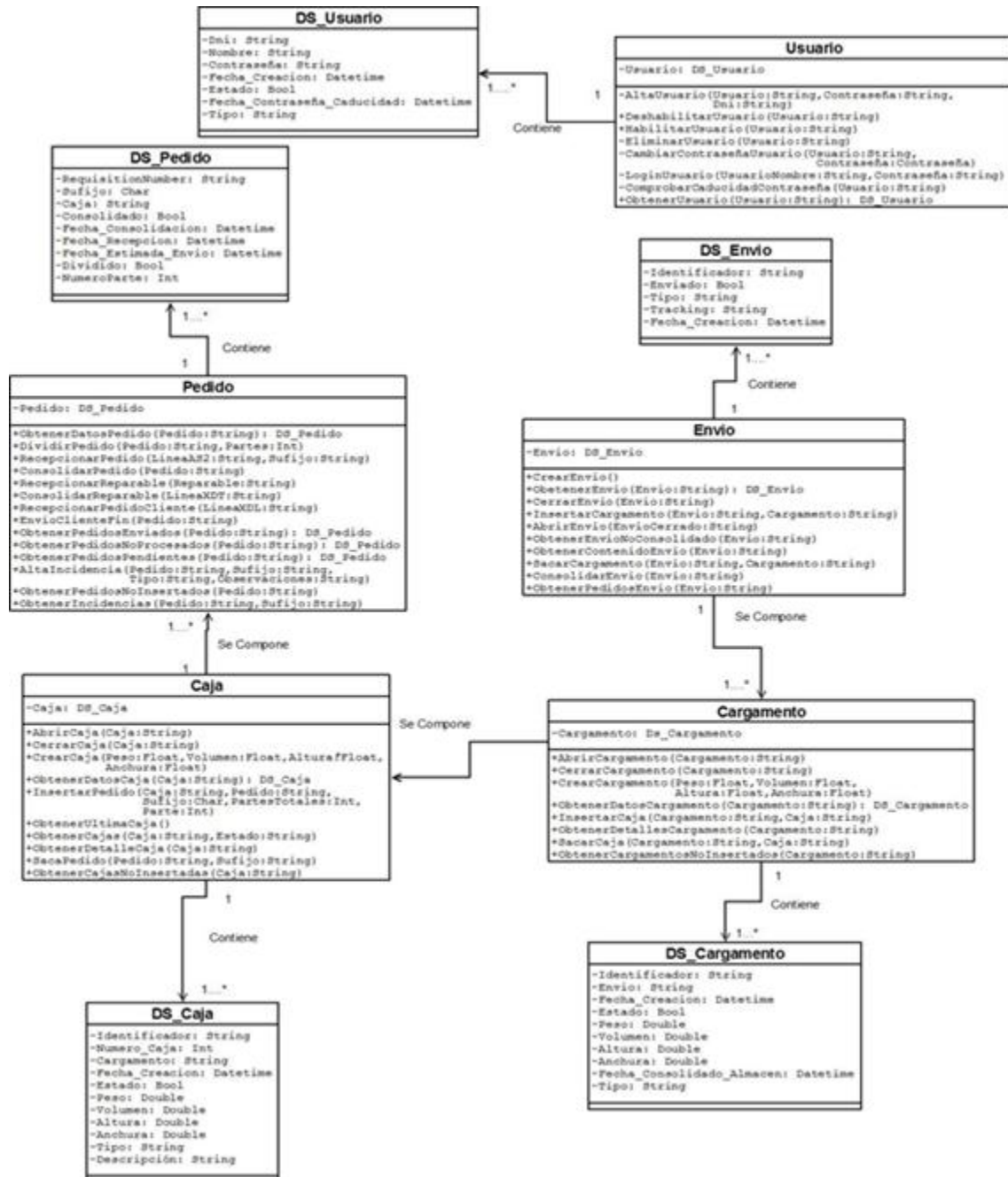
ID: NFSR -44	Tipo: Sistema	Necesidad: Alta
Título: Sesión		
Fuente: Cliente		
Descripción: La sesión del usuario tendrá un tiempo de caducidad.		
Estabilidad: Alta		
Verificabilidad: Medio		
Relación Con Requisitos De Usuario: UR-0025		
Relación Con Requisitos De Sistema: -		

8.3 Arquitectura del sistema

El sistema se implementa en una arquitectura cliente - servidor. En éste apartado se contemplará el modelo de clases con la identificación de todos sus atributos, métodos, y relaciones.

8.3.1 Modelo de clases

38 Diagrama De Clases



8.3.2 Identificación de clases

Las clases identificadas se detallan a continuación:

96 Identificación Clase Pedido

Clase Pedido	
Nombre	Pedido
Tipo	Clase
Descripción	Clase encargada de suministrar a la aplicación los datos de los pedidos junto con todas sus operaciones y procesos

97 Identificación Clase DS_Pedido

Clase DS_Pedido	
Nombre	Pedido
Tipo	Clase-DataSet
Descripción	Clase de tipo DataSet que contiene todos los atributos propios del pedido para suministrar a la clase Pedido.

98 Identificación Clase Caja

Clase Caja	
Nombre	Caja
Tipo	Clase
Descripción	Clase encargada de suministrar a la aplicación los datos de las cajas y métodos apropiados para la gestión de cajas e inserción de pedidos.

99 Identificación Clase DS_Caja

Clase DS_Caja	
Nombre	Caja
Tipo	Clase-DataSet
Descripción	Clase de tipo DataSet que contiene todos los atributos propios de la caja para suministrar a la clase Caja.

100 Identificación Clase Cargamento

Clase Cargamento	
Nombre	Cargamento
Tipo	Clase
Descripción	Clase encargada de suministrar a la aplicación los datos de los cargamentos y métodos apropiados para la gestión de cargamentos e inserción de cajas.

101 Identificación Clase DS_Cargamento

Clase DS_Cargamento	
Nombre	DS_Cargamento
Tipo	Clase-DataSet
Descripción	Clase de tipo DataSet que contiene todos los atributos propios del cargamento para suministrar a la clase Cargamento.

102 Identificación Clase Envío

Clase Envío	
Nombre	Envío
Tipo	Clase
Descripción	Clase encargada de suministrar a la aplicación los datos de los envíos y métodos apropiados para la gestión de envíos e inserción de cargamentos.

103 Identificación Clase DS_Envío

Clase DS_Envío	
Nombre	DS_Envío
Tipo	Clase-DataSet
Descripción	Clase de tipo DataSet que contiene todos los atributos propios del envío para suministrar a la clase Envío.

104 Identificación Clase Usuario

Clase Usuario	
Nombre	Usuario
Tipo	Clase
Descripción	Clase encargada de suministrar a la aplicación los datos de los usuario y métodos apropiados para la gestión de dichos usuarios.

105 Identificación Clase DS_Usuario

Clase DS_Usuario	
Nombre	DS_Usuario
Tipo	Clase-DataSet
Descripción	Clase de tipo DataSet que contiene todos los atributos propios de los usuarios para suministrar a la clase Usuario.

8.2.3 Diseño de Asociaciones y agregaciones

Asociaciones con DataSets:

106 Asociaciones Calce Pedido

Pedido -> DS_Pedido	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	Pedido : DS_Pedido

107 Asociaciones Clase Caja

Caja -> DS_Caja	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	Caja : DS_Caja

108 Asociaciones Clase Cargamento

Cargamento -> DS_ Cargamento	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	Cargamento : DS_ Cargamento

109 Asociaciones Clase Envío

Envío -> DS_ Envío	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	Envío : DS_ Envío

110 Asociación Clase Usuario

Usuario -> DS_ Usuario	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	Usuario : DS_ Usuario

Asociaciones entre clases

111 Asociación Caja - Pedido

Caja -> Pedido	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	InsertarPedido(Caja: String, Pedido: String)

112 Asociación Cargamento Caja

Cargamento -> Caja	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	InsertarCaja(Cargamento: String, Caja: String)

113 Asociación Envío Cargamento

Envío -> Cargamento	
Tipo	Asociación
Navegación	Unidireccional
Clases (Cardinalidad)	1:n
Parámetros - Métodos	InsertarCargamento(Envío: String, Cargamento: String)

8.2.4 Identificación de atributos de las clases

En este apartado se identifican y describen los atributos de las clases de diseño.

Para identificar correctamente estos atributos es necesario revisar el modelo de clases incluyendo sus relaciones.

A continuación se detalla en varias tablas todos los atributos identificados:

114 Atributos Clase Pedido

Clase Pedido	
Pedido	
Visibilidad	Privado
Tipo	DS_Pedido
Descripción	Objeto de tipo DataSet que contiene todos los atributos de un pedido.

115 Atributos Clase DS_Pedido

Clase DS_Pedido	
RequisitionNumber	
Visibilidad	Privado
Tipo	String
Descripción	Identificador del pedido referente a su línea FMS
Sufijo	
Visibilidad	Privado
Tipo	Char
Descripción	Sufijo estándar FMS en caso de que el pedido está dividido
Caja	
Visibilidad	Privado
Tipo	String
Descripción	Caja en la cual está insertado dicho pedido
Consolidado	
Visibilidad	Privado
Tipo	Booleano
Descripción	True si el pedido ha sido consolidado y false en caso contrario
Fecha_Consolidación	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha en la cual ha sido consolidado el pedido
Fecha_Recepción	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha en la cual ha sido recepcionado el pedido.
Fecha_Estimada	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha en la cual se estima que va a ser enviado dicho pedido.
Divisiones Totales	
Visibilidad	Privado
Tipo	Integer
Descripción	Número de divisiones del pedido en caso de tenerlas.
NumeroParte	
Visibilidad	Privado
Tipo	Integer
Descripción	Para controlar el número de particiones del pedido en caso de ser dividido en el almacén.

116 Atributos Clase Caja

Clase Caja	
Caja	
Visibilidad	Privado
Tipo	DS_Caja
Descripción	Objeto de tipo DataSet que contiene todos los atributos de una caja.

117 Atributos Clase DS_Caja

Clase DS_Caja	
Identificador	
Visibilidad	Privado
Tipo	String
Descripción	Identificador propio de la caja
Cargamento	
Visibilidad	Privado
Tipo	String
Descripción	Identificador del cargamento en el cual va insertada la caja.
Fecha_Creacion	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha de creación de la caja.
Estado	
Visibilidad	Privado
Tipo	Booleano
Descripción	True en caso de estar abierta, falso en caso contrario.
Peso	
Visibilidad	Privado
Tipo	Double
Descripción	Peso de la caja.
Volumen	
Visibilidad	Privado
Tipo	Double
Descripción	Volumen de la caja
Altura	
Visibilidad	Privado
Tipo	Double
Descripción	Altura de la caja
Anchura	
Visibilidad	Privado
Tipo	Double
Descripción	Anchura de la caja
Descripción	
Visibilidad	Privado
Tipo	String
Descripción	Breve descripción de la tipología de la caja
Tipo	
Visibilidad	Privado
Tipo	String

Descripción	Tipo de caja.
-------------	---------------

118 Atributos Calce Cargamento

Clase Cargamento	
Cargamento	
Visibilidad	Privado
Tipo	DS_Cargamento
Descripción	Objeto de tipo DataSet que contiene todos los atributos de un cargamento.

119 Atributos Calce DS_Cargamento

Clase DS_Cargamento	
Identificador	
Visibilidad	Privado
Tipo	String
Descripción	Identificador propio del cargamento.
Envío	
Visibilidad	Privado
Tipo	String
Descripción	Envío en el cual está asignado dicho cargamento.
Fecha_Creacion	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha de creación del cargamento.
Estado	
Visibilidad	Privado
Tipo	Booleano
Descripción	True en caso de estar abierto, falso en caso contrario.
Peso	
Visibilidad	Privado
Tipo	Double
Descripción	Peso total del cargamento.
Volumen	
Visibilidad	Privado
Tipo	Double
Descripción	Volumen del cargamento
Altura	
Visibilidad	Privado
Tipo	Double
Descripción	Altura del cargamento
Anchura	
Visibilidad	Privado
Tipo	Double
Descripción	Anchura del cargamento
Fecha_Consolidado_Almacen	
Visibilidad	Privado
Tipo	DateTime

Descripción	Fecha en la cual fue consolidado dicho cargamento.
Tipo	
Visibilidad	Privado
Tipo	String
Descripción	Tipo de cargamento

120 Atributos Clase Envío

Clase Envío	
Envío	
Visibilidad	Privado
Tipo	DS_Envío
Descripción	Objeto de tipo DataSet que contiene todos los atributos de un envío.

121 Atributos Clase DS_Envío

Clase DS_Envío	
Identificador	
Visibilidad	Privado
Tipo	String
Descripción	Identificador propio del envío
Enviado	
Visibilidad	Privado
Tipo	Booleano
Descripción	True en caso de que el envío haya salido del almacén, false en caso contrario.
Tipo	
Visibilidad	Privado
Tipo	String
Descripción	Tipo de envío realizado: por mar, tierra o aire.
Tracking	
Visibilidad	Privado
Tipo	String
Descripción	Tracking generado al ser enviado.

122 Atributos Clase Usuario

Clase Usuario	
Usuario	
Visibilidad	Privado
Tipo	DS_Usuario
Descripción	Objeto de tipo DataSet que contiene todos los atributos de un usuario.

123 Atributos Clase DS_Usuario

Clase DS_Usuario	
Nombre	
Visibilidad	Privado
Tipo	String
Descripción	Nombre del usuario.
Contraseña	
Visibilidad	Privado
Tipo	String
Descripción	Contraseña del usuario
Fecha_Creacion	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha en la cual fue dado de alta el usuario
Estado	
Visibilidad	Privado
Tipo	Booleano
Descripción	True si el usuario está habilitado y false en caso contrario.
Fecha_Contraseña_Caducidad	
Visibilidad	Privado
Tipo	DateTime
Descripción	Fecha en la cual caduca su contraseña
Tipo	
Visibilidad	Privado
Tipo	String
Descripción	Controla el rol del usuario

8.2.5 Identificación y Descripción de las Operaciones

Clase Pedido:

124 Método Clase Pedido ObtenerDatosPedido

ObtenerDatosPedido	
Parámetro	Pedido
Tipo	String
Descripción	Método encargado de consultar la base de datos y devolver los datos del pedido a obtener.

125 Método Clase Pedido DividirPedido

DividirPedido	
Parámetro	Pedido y Partes
Tipo	String e Int
Descripción	Método encargado de dividir el pedido en caso de que sea necesario.

126 Método Clase Pedido RecepcionarPedido

RecepcionarPedido	
Parámetro	LíneaAS2
Tipo	String
Descripción	Método encargado de recepcionar el pedido en el almacén pasándole por parámetro el Identificador de la línea a recepcionar. A continuación se generarán las líneas necesarias bajo el estándar FMS

127 Método Clase Pedido ConsolidarPedido

ConsolidarPedido	
Parámetro	Pedido
Tipo	String
Descripción	Método encargado de consolidar el pedido en el almacén pasándole por parámetro el Identificador de la línea a consolidar. A continuación se generarán las líneas necesarias bajo el estándar FMS

128 Método Clase Pedido RecepcionarReparable

RecepcionarReparable	
Parámetro	LíneaXDR
Tipo	String
Descripción	Método encargado de recepcionar el pedido reparable en el almacén pasándole por parámetro el Identificador de la línea a recepcionar. A continuación se generarán las líneas necesarias bajo el estándar FMS

129 Método Clase Pedido ConsolidarReparable

ConsolidarReparable	
Parámetro	Pedido
Tipo	String
Descripción	Método encargado de consolidar el pedido reparable en el almacén pasándole por parámetro el Identificador de la línea a consolidar. A continuación se generarán las líneas necesarias bajo el estándar FMS

130 Método Clase Pedido RecepcionarPedidoCliente

RecepcionarPedidoCliente	
Parámetro	LíneaXDL
Tipo	String
Descripción	Método encargado de recepcionar el pedido en el almacén del cliente pasándole por parámetro el Identificador de la línea a recepcionar. A continuación se generarán las líneas necesarias bajo el estándar FMS

131 Método Clase Pedido EnvíoClienteFin

EnvíoClienteFin	
Parámetro	Pedido
Tipo	String
Descripción	Método encargado de realizar el último envío del pedido hacia el cliente pasándole por parámetro el Identificador de la línea a enviar. A continuación se generarán las líneas necesarias bajo el estándar FMS

132 ObtenerPedidosEnviados

ObtenerPedidosEnviados	
Parámetro	
Tipo	
Descripción	Método encargado de consultar la base de datos y obtener los pedidos enviados en dicho almacén.

133 ObtenerPedidosNoProcesados

ObtenerPedidosNoProcesados	
Parámetro	
Tipo	
Descripción	Método encargado de consultar la base de datos y obtener los pedidos no procesados en dicho almacén.

134 ObtenerPedidosPendientes

ObtenerPedidosPendientes	
Parámetro	
Tipo	
Descripción	Método encargado de consultar la base de datos y obtener los pedidos pendientes de recepcionar en dicho almacén.

135 Método Clase Pedido AltaIncidencia

AltaIncidencia	
Parámetro	Pedido
Tipo	String
Descripción	Método encargado de dar de alta incidencias en los pedidos.

136 Método Clase Pedido ObtenerPedidosNoInsertados

ObtenerPedidosNoInsertados	
Parámetro	
Tipo	
Descripción	Método encargado de devolver los pedidos que no han sido insertados en cajas.

137 Método Clase Pedido ObtenerIncidencias

ObtenerIncidencias	
Parámetro	Pedido, Sufijo
Tipo	String
Descripción	Método encargado de devolver las incidencias dadas de alta de un pedido.

Clase Caja

138 Método Clase Caja AbrirCaja

AbrirCaja	
Parámetro	Caja
Tipo	String
Descripción	Método encargado de abrir una caja en el almacén cambiando su estado en la base de datos.

139 Método Clase Caja CerrarCaja

CerrarCaja	
Parámetro	Caja
Tipo	String
Descripción	Método encargado de cerrar una caja en el almacén cambiando su estado en

	la base de datos.
--	-------------------

140 Método Clase Caja CrearCaja

CrearCaja	
Parámetro	Peso,Volumen,Altura,Anchura
Tipo	Floats
Descripción	Método encargado de dar de alta una nueva caja en el almacén insertando sus datos en la base de datos.

141 Método Clase Caja ObtenerDatosCaja

ObtenerDatosCaja	
Parámetro	Caja
Tipo	String
Descripción	Método encargado de obtener los datos deseados de una caja

142 Método Clase Caja InsertarPedido

InsertarPedido	
Parámetro	Caja, Pedido, Sufijo
Tipo	String, String, Char
Descripción	Método encargado de insertar el pedido en una caja para su posterior consolidación

143 Método Clase Caja ObtenerUltimaCaja

ObtenerUltimaCaja	
Parámetro	
Tipo	
Descripción	Devuelve el identificador de la última caja creada.

144 Método Clase Caja ObtenerCajas

ObtenerCajas	
Parámetro	Caja, Estado
Tipo	String
Descripción	Según el estado de la caja (Abierta, Cerrada, Todas), devuelve el conjunto de cajas que comiencen por ese identificador.

145 Método Clase Caja ObtenerDetallesCaja

ObtenerDetallesCaja	
Parámetro	Caja
Tipo	String
Descripción	Devuelve la información de la caja pasada por parámetro.

146 Método Clase Caja SacarPedido

SacarPedido	
Parámetro	Pedido, Sufijo, Caja
Tipo	String
Descripción	Elimina el pedido de esa caja pasada por parámetro.

147 Método Clase Caja ObtenerCajasNoInsertadas

ObtenerCajasNoInsertadas	
Parámetro	Caja
Tipo	String
Descripción	Devuelve el conjunto de cajas que no han sido insertadas en un cargamento que comiencen por el identificador del parámetro

Clase Cargamento

148 Método Clase Cargamento AbrirCargamento

AbrirCargamento	
Parámetro	Cargamento
Tipo	String
Descripción	Método encargado de abrir un cargamento en el almacén cambiando su estado en la base de datos.

149 Método Clase Cargamento CerrarCargamento

CerrarCargamento	
Parámetro	Cargamento
Tipo	String
Descripción	Método encargado de cerrar un cargamento en el almacén cambiando su estado en la base de datos.

150 Método Clase Cargamento CrearCargamento

CrearCargamento	
Parámetro	Peso,Volumen,Altura,Anchura
Tipo	Floats
Descripción	Método encargado de dar de alta un nuevo cargamento en el almacén insertando sus datos en la base de datos.

151 Método Clase Cargamento ObtenerDatosCargamento

ObtenerDatosCargamento	
Parámetro	Cargamento
Tipo	String
Descripción	Método encargado de obtener los datos deseados de un cargamento

152 Método Clase Cargamento InsertarCaja

InsertarCaja	
Parámetro	Cargamento, Caja
Tipo	String, String
Descripción	Método encargado de insertar una caja en un cargamento para su posterior consolidación

153 Método Clase Cargamento ObtenerDetallesCargamento

ObtenerDetallesCargamento	
Parámetro	Cargamento
Tipo	String
Descripción	Devuelve la información y detalles del cargamento pasado por parámetro.

154 Método Clase Cargamento SacarCaja

SacarCaja	
Parámetro	Cargamento, Caja
Tipo	String
Descripción	Método encargado de sacar cajas de un cargamento.

155 Método Clase Cargamento ObtenerCargamentosNoInsertados

ObtenerCargamentosNoInsertados	
Parámetro	Cargamento
Tipo	String
Descripción	Devuelve el conjunto de cargamentos no insertados. Según el parámetro se devolverán los cargamentos que empiecen por ese nombre.

Clase Envío

156 Método Clase Envío CrearEnvío

CrearEnvío	
Parámetro	
Tipo	
Descripción	Método encargado de crear un nuevo envío asignándole sus identificadores por defecto.

157 Método Clase Envío ObtenerEnvío

ObtenerEnvío	
Parámetro	Envío
Tipo	String
Descripción	Método encargado de obtener los datos deseados de un envío

158 Método Clase Envío InsertarCargamento

InsertarCargamento	
Parámetro	Envío, Cargamento
Tipo	String, String
Descripción	Método encargado de insertar un cargamento en dicho envío.

159 Método Clase Envío CerrarEnvío

CerrarEnvío	
Parámetro	Envío
Tipo	String
Descripción	Método encargado de cerrar un envío en el almacén cambiando su estado en la base de datos y generando las líneas FMS necesarias

160 Método Clase Envío ObtenerEnvíoNoConsolidado

ObtenerEnvíoNoConsolidado	
Parámetro	Envío
Tipo	String
Descripción	Devuelve el conjunto de envíos no consolidados que comiencen por el identificador del parámetro.

161 Método Clase Envío ObtenerContenidoEnvío

ObtenerContenidoEnvío	
Parámetro	Envío
Tipo	String
Descripción	Método encargado de obtener los cargamentos que forman parte del envío pasado por parámetro.

162 Método Clase Envío SacarCargamento

SacarCargamento	
Parámetro	Envío, Cargamento
Tipo	String
Descripción	Método encargado de eliminar el cargamento asociado a dicho envío.

163 Método Clase Envío ConsolidarEnvío

ConsolidarEnvío	
Parámetro	Envío
Tipo	String
Descripción	Método encargado de consolidar el envío , creando las líneas necesarias por pedidos contenidos en dicho envío.

164 Método Clase Envío ObtenerPedidosEnvío

ObtenerPedidosEnvío	
Parámetro	Envío
Tipo	String
Descripción	Método encargado de devolver todos los pedidos que contienen dicho envío.

Clase Usuario

165 Método Clase Usuario AltaUsuario

AltaUsuario	
Parámetro	Datos Usuario
Tipo	Valores Del Usuario
Descripción	Método encargado de dar de alta un nuevo usuario en el sistema

166 Método Clase Usuario DeshabilitarUsuario

DeshabilitarUsuario	
Parámetro	Usuario
Tipo	String
Descripción	Método encargado de deshabilitar un usuario cambiando su estado en la base de datos

167 Método Clase Usuario EliminarUsuario

EliminarUsuario	
Parámetro	Usuario
Tipo	String
Descripción	Método encargado de eliminar un usuario suprimiendo su registro en la base de datos

168 Método Clase Usuario CambiarContraseñaUsuario

CambiarContraseñaUsuario	
Parámetro	Usuario, Contraseña
Tipo	String, String
Descripción	Método encargado de cambiar la contraseña del usuario pasándole la nueva contraseña a actualizar en la base de datos.

169 Método Clase Usuario LoginUsuario

LoginUsuario	
Parámetro	Usuario, Contraseña
Tipo	String, String
Descripción	Método encargado de loguear el usuario en la aplicación

170 Método Clase Usuario ComprobarCaducidadContraseña

ComprobarCaducidadContraseña	
Parámetro	Usuario, Contraseña
Tipo	String, String
Descripción	Método encargado de comprobar que la contraseña del usuario no ha caducado.

171 Método Clase Usuario ObtenerDatosUsuario

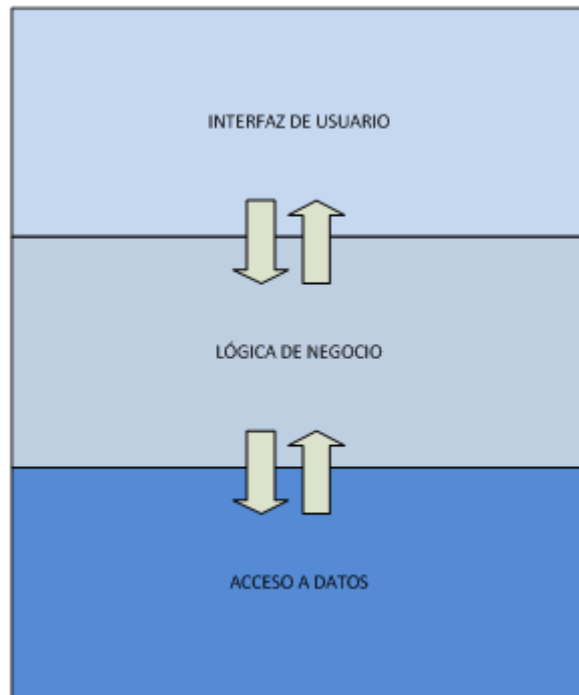
ObtenerDatosUsuario	
Parámetro	Usuario
Tipo	String
Descripción	Método encargado de obtener los datos del usuario.

8.4 Diseño de la arquitectura de módulos del sistema

8.4.1 Diseño de Módulos del Sistema y comunicación entre ellos

Se ha llevado a cabo una arquitectura en 3 capas.

Así obtenemos una separación entre la capa de presentación (Interfaz de Usuario), la capa de negocio (Lógica de Negocio) y la capa de datos (Acceso A Datos).



39 Desarrollo De La Arquitectura En 3 Capas

1. Capa De Presentación o Interfaz De Usuario: Esta capa es la que ve el usuario, presenta el sistema al usuario, le comunica y captura la información del usuario en un mínimo de procesos. Esta capa se comunica únicamente con la capa de negocio.
2. Capa De Negocio o Lógica De Negocio: Aquí es donde, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación para recibir las solicitudes y presentar los resultados y con la capa de datos para solicitar al gestor de base de datos la información

necesaria o en caso contrario guardar nueva información en la base de datos.

3. Capa de Datos o Acceso A Datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. En dicho proyecto el gestor de base de datos es *Sql Server 2008*.

8.5 Diseño físico de datos

8.5.1 Diseño del Modelo Físico de Datos

8.4.4.1 Diccionario de datos

La especificación llevará el siguiente formato:

Nombre Tabla	
Campo	Tipo

172 Diccionario Tabla Caja

CAJA	
Campo	Tipo
Identificador	Nvarchar(20)
Cargamento	Nvarchar(20)
Fecha_Creación	Datetime
Estado	Bit
Peso	Float
Volumen	Float
Altura	Float
Anchura	Float
Tipo	Nvarchar(20)
Descripción	Nvarchar(150)

173 Dicionario Tabla Cargamento

CARGAMENTO	
Campo	Tipo
Identificador	Nvarchar(20)
Envió	Nvarchar(20)
Fecha_Creación	Datetime
Estado	Bit
Peso	Float
Volumen	Float
Altura	Float
Anchura	Float
Tipo	Nvarchar(20)

174 Dicionario Tabla Envió

ENVIO	
Campo	Tipo
Identificador	Nvarchar(20)
Enviado	Bit
Tipo	Nvarchar(20)
Tracking	Nvarchar(50)
Fecha_Creacion	Datetime

175 Dicionario Tabla Incidencias

INCIDENCIAS	
Campo	Tipo
Numero	Nvarchar(20)
Pedido	Nvarchar(13)
Sufijo	Nvarchar(2)
Tipo	Nvarchar(20)
Observaciones	Nvarchar(150)
Usuario	Nvarchar(20)
Fecha_Creacion	Datetime

176 Dicionario Tabla Línea

LINEA	
Campo	Tipo
Identificador	Nvarchar(4)
Numero_Documento	Nvarchar(13)
Línea	Nvarchar(300)
Fecha_Entrada	Bit
Sufijo	Nvarchar(2)

177 Diccionario Tabla Pedido

Pedido	
Campo	Tipo
Numero_Documento	Nvarchar(13)
Sufijo	Nvarchar(2)
Consolidado	Bit
Fecha_Consolidación	Datetime
Fecha_Recepcion	Datetime
Fecha_Estimada	Datetime

178 Diccionario Tabla Pedido_Caja

Pedido_Caja	
Campo	Tipo
Caja	Nvarchar(20)
Pedido	Nvarchar(13)
Sufijo	Nvarchar(2)
Fecha_Inserción	Datetime
Divisiones_Totales	Integer
Numero_División	Integer

179 Diccionario Tabla Usuarios

Usuarios	
Campo	Tipo
Dni	Nvarchar(9)
Nombre	Nvarchar(20)
Contraseña	Nvarchar(50)
Fecha_Creación	Datetime
Estado	Bit
Tipo	Nvarchar(15)
Fecha_Caducidad	Datetime

8.5.2 Especificación del Script De Creación Almacén EE.UU

Sera necesario definir dos bases de datos, una residente en EEUU y que almacene los datos del envío y otra en un lugar local que almacene la información de los procesos....

Ambas bases de datos están sincronizadas y las restricciones de integridad referencial aseguran su cohesión de tal forma que garantizamos que todas las asociaciones son válidas.

El siguiente esquema muestra cómo se establecen dichas bases de datos:



40 Comunicación Bases De Datos Aplicación

Los scripts se localizan en los anexos del final del documento.

8.6 Diseño De La Interfaz

El diseño de las interfaces de usuario se estructura de la siguiente forma:



Ilustración 1 Diseño De Las Interfaces

El menú superior sirve para poder navegar por todas las secciones de la aplicación.

El buscador sirve para buscar cualquier tipo de pedido, caja, cargamento o envío obteniendo la información deseada.

La última parte se compone de los Grids para visualizar el contenido de la búsqueda antes detallada, de tal forma que se pueda interactuar con los elementos mencionados anteriormente.

A continuación se muestran algunas de las interfaces de la aplicación, de tal forma que resumen el diseño de toda ella.

Interfaz Login:



41 Interfaz De Login

Interfaz Alta Usuario:



42 Interfaz Alta Usuario

Interfaz Cambio De Password:



Universidad Carlos III de Madrid

CAMBIAR PASSWORD

Usuario: adri2
Dni: 50543966V
Contraseña:
Nueva Contraseña:
Repetir Nueva Contraseña:


**** El campo Password debe tener:**
Al menos un caracter en mayusculas.
Uno en minusculas.
Uno numerico
Un minimo 8 caracteres de longitud
Caracteres especiales se admitidos @#\$%^&+=_

Universidad Carlos III |

43 Interfaz Cambio De Password

Interfaz Búsqueda Pedidos

44 Interfaz Búsqueda De Pedidos



Universidad Carlos III de Madrid

Principal Gestion De Caja Gestion De Cargamento Gestion De Envios Alta Incidencia Recepciones Consultas FMS Consolidación Salir

Buscar Pedido
Tipo De Pedido: Enviados
Identificador:

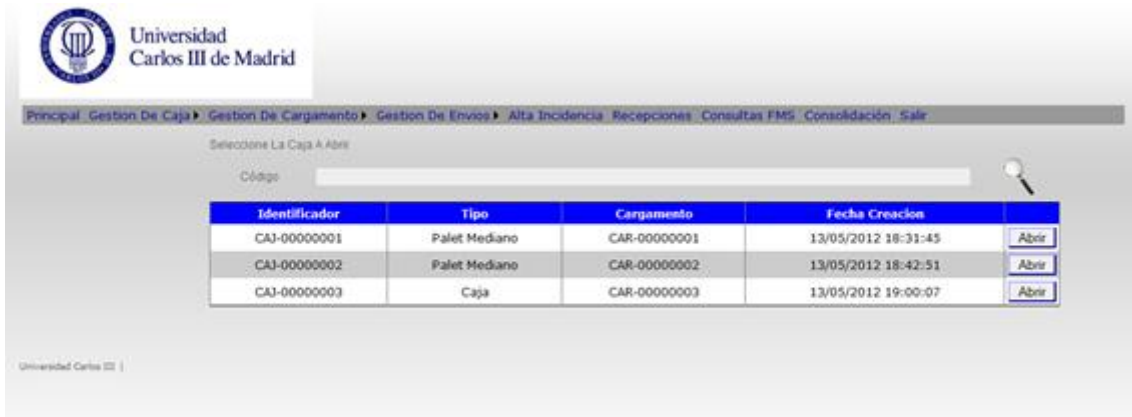
Pedido	Sufijo	Fecha De Envio
pedido1	0	13/05/2012 18:39:16
Pedido2	0	13/05/2012 18:43:26
Reparable	0	13/05/2012 19:00:41

Universidad Carlos III |

Secuencia de inserción de pedido en una caja previamente cerrada.

1. Se abre la caja previamente cerrada:

Interfaz Abrir Caja



Universidad Carlos III de Madrid

Principal | Gestion De Caja | Gestion De Cargamento | Gestion De Envios | Alta Incidencia | Recepciones | Consultas FMS | Consolidación | Salir

Seleccione La Caja A Abrir

Código:

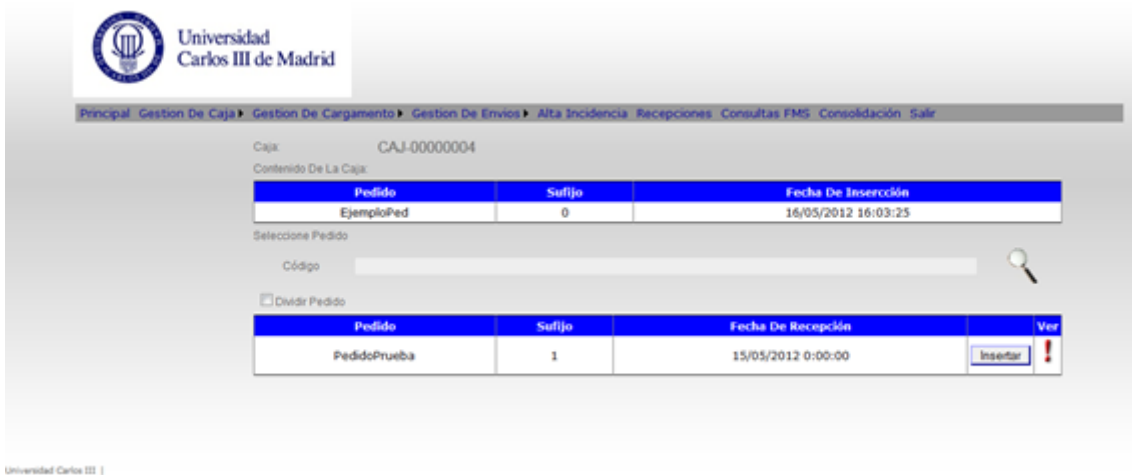
Identificador	Tipo	Cargamento	Fecha Creacion	
CAJ-00000001	Palet Mediano	CAR-00000001	13/05/2012 18:31:45	<input type="button" value="Abrir"/>
CAJ-00000002	Palet Mediano	CAR-00000002	13/05/2012 18:42:51	<input type="button" value="Abrir"/>
CAJ-00000003	Caja	CAR-00000003	13/05/2012 19:00:07	<input type="button" value="Abrir"/>

Universidad Carlos III |

45 Interfaz Abrir Caja

2. Se selecciona la caja, se busca el pedido en el buscador y se inserta pulsando el botón de insertar.

Interfaz Insertar Pedido



Universidad Carlos III de Madrid

Principal | Gestion De Caja | Gestion De Cargamento | Gestion De Envios | Alta Incidencia | Recepciones | Consultas FMS | Consolidación | Salir

Caja: CAJ-00000004

Contenido De La Caja:

Pedido	Sufijo	Fecha De Inserción
EjemploPed	0	16/05/2012 16:03:25

Seleccione Pedido

Código:

☐ Dividir Pedido

Pedido	Sufijo	Fecha De Recepción		Ver
PedidoPrueba	1	15/05/2012 0:00:00	<input type="button" value="Insertar"/>	<input type="button" value="Ver"/>

Universidad Carlos III |

46 Interfaz Insertar Pedido

3. Se cierra la caja. Una vez cerrada ya está lista para insertarla en un cargamento.

Interfaz Cerrar Caja

Principal Gestion De Caja Gestion De Cargamento Gestion De Envios Alta Incidencia Recepciones Consultas FMS Consolidación Salir

Seleccione La Caja A Cerrar:

Código:

Identificador	Tipo	Cargamento	Fecha Creacion	Cerrar
CAJ-00000004	Prueba		14/05/2012 19:09:17	Cerrar
CAJ-00000007	Palet Mediano		16/05/2012 16:02:26	Cerrar
CAJ-00000008	Cajon		16/05/2012 16:02:28	Cerrar
CAJ-00000009	Palet Grande		16/05/2012 16:02:29	Cerrar

Universidad Carlos III

47 Interfaz Cerrar Caja

Interfaz Detalle Cargamento

Principal Gestion De Caja Gestion De Cargamento Gestion De Envios Alta Incidencia Recepciones Consultas FMS Consolidación Salir

Seleccione El Cargamento Para Ver Sus Detalles:

Código:

Identificador	Estado	Envío	Tipo	Fecha Creacion	Seleccionar
CAR-00000001	False	ENV-00000001		13/05/2012 18:38:51	Seleccionar
CAR-00000002	False	ENV-00000002		13/05/2012 18:43:03	Seleccionar
CAR-00000003	False	ENV-00000003		13/05/2012 19:00:19	Seleccionar
CAR-00000004	False	ENV-00000004	Contenedor Prueba	16/05/2012 16:06:39	Seleccionar

Cargamento: CAR-00000004
 Fecha De Creación: 16/05/2012 16:06:39
 Envío Asociado: ENV-00000004
 Estado: Cerrado
 Tipo: Contenedor Prueba
 Peso: 125
 Volumen: 12
 Altura: 1500
 Anchura: 1800

Cajas Del Cargamento:

Caja	Ver
CAJ-00000004	<input type="button" value="Sacar"/>

48 Interfaz Detalle Cargamento

Ejemplo Del Menú Superior

Principal Gestion De Caja Gestion De Cargamento Gestion De Envios Alta Incidencia Recepciones Consultas FMS Consolidación Salir

Seleccione El Cargamento A Cerrar:

Código:

Crear Envío
 Asignar Cargamentos
 Detalles Envío

49 Interfaz Menu Superior

8.7 Alternativas de Diseño

8.7.1 Gestor De Base De Datos

El sistema Gestor de Base de Datos escogido es el Microsoft Sql Server 2008.

Se han barajado varias alternativas:

- Mysql: además es un gestor completamente gratuito.
- Oracle: similar respecto al funcionamiento con Sql Server 2008

Mysql aun siendo gratuito se ha desechado por el simple hecho de que es un gestor con muy poca funcionalidad de administración y gestión de usuarios. Dicha aplicación necesita obligatoriamente unos sistemas de seguridad muy estrictos al estar tratando con mercancías confidenciales.

Oracle a diferencia de Mysql es un gestor de pago como Sql Server incluyendo funcionalidades de administración, gestión de usuarios y de seguridad y permisos. Oracle no ha sido elegido por el simple hecho de que ésta aplicación va a ser ejecutada sobre el sistema operativo de Windows enlazando al gestor de base de datos sobre el servidor Internet Information Services de Microsoft.

Por tanto la mejor elección es usar Microsoft Sql Server.

[Más información respecto a Sql Server 2008 y diferencias respecto a otras en la sección de

Gestión del Proyecto -> Herramientas -> Sql Server 2008]

8.7.2 Sistema Operativo

El Sistema Operativo elegido es Windows.

Otras opciones es usar una distribución de Linux como Ubuntu o Red Hat ó el sistema operativo Mac OS X de Apple.

La elección de Windows, es por la misma razón del sistema gestor de base de datos. Tanto el sistema operativo como el gestor de base de datos son propios de Microsoft, por lo que la administración y gestión de ambos es mucho más sencilla que realizar una mezcla de ambos.

8.7.3 Aplicación

La aplicación es una aplicación Web.

Otra opción sería realizar una aplicación de escritorio.

El problema que se ha encontrado al estudiar realizar una aplicación de escritorio (tal como son las estudiadas en el apartado Estado Del Arte o Estado De La Cuestión), es el acceso a ellas y su instalación.

Al ser una aplicación Web es mucho más simple el hecho de acceso a ella, ya que cualquier ordenador con acceso a internet puede conectar con la aplicación y su base de datos, incluso desde un dispositivo móvil.

8.7.4 Interfaces De La Aplicación

La aplicación está desarrollada bajo el uso de Grids por la mayoría de sus secciones, de ésta forma el usuario obtiene una aplicación visualmente muy atractiva, con un diseño muy definido y flexible.

Con el uso del Framework de desarrollo Visual Studio 2010 es posible llevar a cabo dicho diseño e implementación.

[Más información respecto a Visual Studio 2010 y diferencias respecto a otras en la sección de

Gestión del Proyecto -> Herramientas -> Microsof Visual Studio 2010]

9 Documento De Pruebas

9.1 Especificación De Pruebas Unitarias

Las pruebas realizadas serán divididas en los siguientes niveles mostrados a continuación:

Pruebas Unitarias:

- **Descripción:** Conjunto de pruebas que tienen como finalidad probar el correcto funcionamiento de un componente, sistema o subsistema. Esto sirve para asegurar que cada uno de los componentes funcione correctamente por separado. La idea es escribir casos de prueba para cada función no trivial o método en el componente de forma que cada caso sea independiente del resto. Hay dos tipos:

Pruebas de Caja Blanca: Tipo de pruebas unitarias que se realiza sobre las funciones o métodos internos de un subsistema o componente.

Pruebas de Caja Negra: Tipos de pruebas unitarias que se realizan sobre los componentes o subsistemas comprobando las entradas que recibe y las salidas o respuestas que produce.

Dichas pruebas serán realizadas a nivel de desarrollo e implementación.

9.1.1 Pruebas De Caja Blanca

180 Prueba Caja Blanca 01

PR-CB-01	
Objetivo	Comprobar que un usuario ha sido dado de alta correctamente.
Clase	Usuario
Método	AltaUsuario
Entrada	Recibe el nombre y la contraseña del usuario a añadir
Salida	Texto informando de que el usuario ha sido dado de alta correctamente. Alternativas: <ul style="list-style-type: none">- Error: No se han rellenado los campos necesarios.- Error: Fallo en la conexión durante la inserción.

181 Prueba Caja Blanca 02

PR-CB-02	
Objetivo	Comprobar que un usuario ha sido deshabilitado correctamente.
Clase	Usuario
Método	DeshabilitarUsuario
Entrada	Recibe el nombre del usuario a deshabilitar.
Salida	<p>Texto informando de que el usuario ha sido dado de alta correctamente.</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: No se ha seleccionado ningún usuario. - Error: Fallo en la conexión durante la actualización.

182 Prueba Caja Blanca 03

PR-CB-03	
Objetivo	Comprobar que un usuario ha sido eliminado de la base de datos
Clase	Usuario
Método	EliminarUsuario
Entrada	Recibe el nombre del usuario a eliminar.
Salida	<p>Texto informando de que el usuario ha sido eliminado correctamente.</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: No se ha seleccionado ningún usuario a eliminar. - Error: Fallo en la conexión durante la supresión.

183 Prueba Caja Blanca 04

PR-CB-04	
Objetivo	Comprobar que se ha cambiado correctamente la contraseña del usuario
Clase	Usuario
Método	CambiarContraseñaUsuario
Entrada	Recibe el nombre del usuario y su contraseña a cambiar.
Salida	<p>Texto informando de que la contraseña se ha cambiado correctamente.</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: No se han rellenado el campo de la contraseña. - Error: Fallo en la conexión durante la actualización.

184 Prueba Caja Blanca 05

PR-CB-05	
Objetivo	Comprobar que un usuario puede loguearse correctamente en la aplicación
Clase	Usuario
Método	LoginUsuario
Entrada	Recibe el nombre y la contraseña del usuario que desea entrar en la aplicación.
Salida	<p>Interfaz de inicio de la aplicación.</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: No se han rellenado los campos de usuario o contraseña - Error: Fallo en la conexión .

185 Prueba Caja Blanca 06

PR-CB-06	
Objetivo	Comprobar que la contraseña de un usuario ha caducado.
Clase	Usuario
Método	ComprobarCaducidad
Entrada	Recibe el nombre del usuario que se desea comprobar
Salida	Texto informando de que su contraseña ha caducado. Alternativas: <ul style="list-style-type: none"> - Error: No se ha seleccionado ningún usuario a comprobar - Error: Fallo en la conexión.

186 Prueba Caja Blanca 07

PR-CB-07	
Objetivo	Comprobar que un usuario ha sido eliminado de la base de datos
Clase	Usuario
Método	EliminarUsuario
Entrada	Recibe el nombre del usuario a eliminar.
Salida	Texto informando de que el usuario ha sido eliminado correctamente. Alternativas: <ul style="list-style-type: none"> - Error: No se ha seleccionado ningún usuario a eliminar. - Error: Fallo en la conexión durante la supresión.

187 Prueba Caja Blanca 08

PR-CB-08	
Objetivo	Comprobar que un cargamento se abre correctamente
Clase	Cargamento
Método	AbrirCargamento
Entrada	Recibe el nombre del cargamento.
Salida	Datos del contenido del cargamento ya abierto. Alternativas: <ul style="list-style-type: none"> - Error: El cargamento pertenece a un envío ya consolidado. - Error: Fallo en la conexión. - Error: No se ha seleccionado ningún cargamento para abrir.

188 Prueba Caja Blanca 09

PR-CB-09	
Objetivo	Comprobar que un cargamento se cierra correctamente
Clase	Cargamento
Método	CerrarCargamento
Entrada	Recibe el nombre del cargamento.
Salida	Cargamento cerrado correctamente Alternativas: <ul style="list-style-type: none"> - Error: El contenido del cargamento no incluye ninguna caja. - Error: El contenido del cargamento incluye cajas abiertas. - Error: Fallo en la conexión. - Error: No se ha seleccionado ningún cargamento para abrir.

189 Prueba Caja Blanca 10

PR-CB-10	
Objetivo	Comprobar que un nuevo cargamento se ha creado correctamente.
Clase	Cargamento
Método	CrearCargamento
Entrada	Recibe los datos del nuevo cargamento a crear
Salida	Cargamento cerrado correctamente Alternativas: <ul style="list-style-type: none"> - Error: Los datos introducidos no son válidos. - Error: No se han rellenado los campos obligatorios - Error: Fallo en la conexión.

190 Prueba Caja Blanca 11

PR-CB-11	
Objetivo	Insertar una caja en un cargamento
Clase	Cargamento
Método	InsertarCaja
Entrada	Recibe el cargamento y la caja a insertar
Salida	Caja insertada correctamente Alternativas: <ul style="list-style-type: none"> - Error: La caja no está cerrada - Error: El cargamento no está abierto. - Error: El cargamento está lleno. - Error: Fallo en la conexión.

191 Prueba Caja Blanca 12

PR-CB-12	
Objetivo	Crear nuevo envío
Clase	Envío
Método	CrearEnvio
Entrada	Recibe los datos del envío a crear
Salida	Envío creado correctamente Alternativas: <ul style="list-style-type: none"> - Error: Los datos del envío no son válidos - Error: Faltan datos obligatorios del envío. - Error: Fallo en la conexión.

192 Prueba Caja Blanca 13

PR-CB-13	
Objetivo	Obtener envío
Clase	Envio
Método	ObtenerEnvio
Entrada	Recibe el envío a obtener
Salida	Datos del envío con el contenido de sus cargamentos Alternativas: <ul style="list-style-type: none"> - Error: El envío no existe

	<ul style="list-style-type: none"> - Error: Falta el campo del nombre del envío a rellenar - Error: Fallo en la conexión.
--	---

193 Prueba Caja Blanca 14

PR-CB-14	
Objetivo	Cerrar envío
Clase	Envío
Método	CerrarEnvío
Entrada	Recibe el envío a cerrar
Salida	Envío consolidado correctamente Alternativas: <ul style="list-style-type: none"> - Error: El envío no ha sido seleccionado. - Error: El envío contiene cargamentos abiertos. - Error: Fallo en la conexión.

194 Prueba Caja Blanca 15

PR-CB-15	
Objetivo	Insertar Cargamento en envío
Clase	Envío
Método	InsertarCargamento
Entrada	Recibe el cargamento y el envío donde se quiere insertar
Salida	Cargamento insertado correctamente Alternativas: <ul style="list-style-type: none"> - Error: El cargamento no está cerrado. - Error: El envío está lleno. - Error: El cargamento no contiene cajas en su interior. - Error: Fallo en la conexión.

195 Prueba Caja Blanca 16

PR-CB-16	
Objetivo	Abrir un envío existente.
Clase	Envío
Método	AbrirEnvío
Entrada	Recibe el envío que se quiere abrir
Salida	Envío abierto correctamente Alternativas: <ul style="list-style-type: none"> - Error: El envío no ha sido seleccionado. - Error: El envío ha sido consolidado y no es posible su apertura. - Error: Fallo en la conexión.

196 Prueba Caja Blanca 17

PR-CB-17	
Objetivo	Abrir caja existente
Clase	Caja
Método	AbrirCaja
Entrada	Recibe la caja que se quiere abrir
Salida	<p>Caja abierta correctamente</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: La caja no ha sido seleccionada. - Error: La caja no se puede abrir al pertenecer a un cargamento cerrado. - Error: Fallo en la conexión.

197 Prueba Caja Blanca 18

PR-CB-18	
Objetivo	Cerrar caja existente
Clase	Caja
Método	CerrarCaja
Entrada	Recibe la caja que se quiere cerrar
Salida	<p>Caja cerrada correctamente</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: La caja no ha sido seleccionada. - Error: La caja no se puede cerrar al no contener ningún pedido en su contenido.. - Error: Fallo en la conexión.

198 Prueba Caja Blanca 19

PR-CB-19	
Objetivo	Crear nueva caja
Clase	Caja
Método	CrearCaja
Entrada	Recibe los datos de la caja a crear.
Salida	<p>Caja creada correctamente</p> <p>Alternativas:</p> <ul style="list-style-type: none"> - Error: Los datos introducidos no son válidos. - Error: Faltan datos obligatorios para la creación de la caja. - Error: Fallo en la conexión.

199 Prueba Caja Blanca 20

PR-CB-20	
Objetivo	Obtener datos de una caja existente
Clase	Caja
Método	ObtenerDatosCaja
Entrada	Recibe la caja de la cual se quiere obtener la información
Salida	Datos de la caja. Alternativas: <ul style="list-style-type: none"> - Error: La caja no ha sido seleccionada. - Error: La caja no existe. - Error: Fallo en la conexión.

200 Prueba Caja Blanca 21

PR-CB-21	
Objetivo	Insertar Pedido en caja
Clase	Caja
Método	InsertarPedido
Entrada	Recibe la caja y el pedido a insertar en ella.
Salida	Pedido Insertado Correctamente Alternativas: <ul style="list-style-type: none"> - Error: La caja no permite la inserción del pedido al sobrepasar los atributos de la caja. - Error: La caja no ha sido seleccionada. - Error: El pedido no ha sido seleccionado. - Error: Fallo en la conexión.

201 Prueba Caja Blanca 22

PR-CB-22	
Objetivo	Obtener datos de un pedido recepcionado
Clase	Pedido
Método	ObtenerDatosPedido
Entrada	Recibe el pedido del que se desea la información.
Salida	Información del pedido Alternativas: <ul style="list-style-type: none"> - Error: El pedido no existe o no ha sido recepcionado en el almacén. - Error: No ha sido seleccionado el pedido. - Error: Fallo en la conexión.

202 Prueba Caja Blanca 23

PR-CB-23	
Objetivo	Dividir un pedido para su posterior consolidación
Clase	Pedido
Método	DividirPedido
Entrada	Recibe el pedido del que se desea particionar y el número de partes lógicas.
Salida	Pedido particionado Alternativas: <ul style="list-style-type: none"> - Error: El pedido no existe o no ha sido recepcionado en el almacén. - Error: No ha sido seleccionado el pedido. - Error: Fallo en la conexión.

203 Prueba Caja Blanca 24

PR-CB-24	
Objetivo	Recepcionar un pedido en el almacén.
Clase	Pedido
Método	RecepcionarPedido, RecepcionarReparable, RecepcionarPedidoCliente
Entrada	Recibe el identificador del pedido y actualiza la base de datos.
Salida	Pedido Recepcionado Correctamente Alternativas: <ul style="list-style-type: none"> - Error: No existe la línea que pertenece a dicho pedido. - Error: Datos no válidos del pedido. - Error: No se ha rellenado el campo del pedido a buscar. - Error: Fallo en la conexión.

204 Prueba Caja Blanca 25

PR-CB-25	
Objetivo	Consolidar y posibilitar un envío.
Clase	Pedido
Método	ConsolidarPedido, ConsolidarReparable, EnvioClienteFin
Entrada	Recibe el identificador del pedido y actualiza la base de datos.
Salida	Pedido Consolidado Correctamente Alternativas: <ul style="list-style-type: none"> - Error: No existe la línea que pertenece a dicho pedido. - Error: Datos no válidos del pedido. - Error: No se ha rellenado el campo del pedido a consolidar. - Error: Fallo en la conexión. - Error: No se ha podido generar las líneas por ausencia de campos obligatorios.

205 Prueba Caja Blanca 26

PR-CB-26	
Objetivo	Obtener información de un pedido.
Clase	Pedido
Método	ObtenerPedidosEnviados,ObtenerPedidosNoProcesados,ObtenerPedidosPendientes
Entrada	
Salida	Lista de pedidos. Alternativas: - Error: Fallo en la conexión.

206 Prueba Caja Blanca 27

PR-CB-27	
Objetivo	Añadir una incidencia a un pedido
Clase	Pedido
Método	AltaIncidencia
Entrada	Pedido con los datos de su incidencia
Salida	Incidencia dada de alta. Alternativas: - Error: No se ha seleccionado pedido para dar de alta incidencia. - Error: Fallo en la validación de datos de la incidencia. - Error: Ausencia de campos obligatorios en la incidencia. - Error: Fallo en la conexión.

9.1.2 Pruebas De Caja Negra

207 Prueba Caja Negra 01

PR-CN-01	
Objetivo	Comprobar que un usuario ha sido dado de alta correctamente.
Clase	Usuario
Método	AltaUsuario
Entrada	Recibe el nombre y la contraseña del usuario a añadir
Descripción	Se llama al método AltaUsuario pasándole por parámetro el nombre y la contraseña del nuevo usuario a añadir. A continuación se inserta el nuevo usuario en la base de datos estableciendo sus campos y añadiendo la fecha en la que se ha insertado.
Salida	Se devuelve el número de líneas que se han añadido, las cuales deben ser de 1. Alternativas (Se devuelve el valor 0 o una excepción): - Error: No se han añadido los parámetros necesarios al método. - Error: Los valores de los parámetros no son correctos. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -01 y FSR -02

208 Prueba Caja Negra 02

PR-CN-02	
Objetivo	Comprobar que un usuario ha sido deshabilitado correctamente.
Clase	Usuario
Método	DeshabilitarUsuario
Entrada	Recibe el nombre del usuario a deshabilitar.
Descripción	Se llama al método DeshabilitarUsuario pasándole por parámetro el nombre del usuario a deshabilitar. A continuación se actualiza el usuario en la base de datos.
Salida	Se devuelve el número de líneas que se han modificado, las cuales deben ser de 1. Alternativas (Se devuelve el valor 0 o una excepción): Error: No se han añadido el parámetro necesario al método. Error: El valor del parámetro no es correcto. Error: Fallo técnico en la base de datos. Error: Usuario no encontrado en la base de datos.
Requisito Software Asociado	FSR –01

209 Prueba Caja Negra 03

PR-CN-03	
Objetivo	Comprobar que un usuario ha sido eliminado de la base de datos
Clase	Usuario
Método	EliminarUsuario
Entrada	Recibe el nombre del usuario a eliminar.
Descripción	Se llama al método EliminarUsuario pasándole por parámetro el nombre del usuario que se quiere eliminar. A continuación se actualiza la base de datos eliminando la línea de dicho usuario.
Salida	Se devuelve el número de líneas que se han eliminado, las cuales deben ser de 1. Alternativas (Se devuelve el valor 0 o una excepción): <ul style="list-style-type: none"> - Error: No se han añadido el parámetro necesario al método. - Error: El valor del parámetro no es correcto. - Error: Fallo técnico en la base de datos. - Error: Usuario no encontrado en la base de datos.
Requisito Software Asociado	FSR –01

210 Prueba Caja Negra 04

PR-CN-04	
Objetivo	Comprobar que se ha cambiado correctamente la contraseña del usuario
Clase	Usuario
Método	CambiarContraseñaUsuario
Entrada	Recibe el nombre del usuario y su contraseña a cambiar.
Descripción.	Se llama al método CambiarContraseñaUsuario pasándole el nombre del usuario y su contraseña a modificar. A continuación se actualiza la base de datos actualizando la contraseña de dicho usuario.
Salida	Se devuelve el número de líneas que se han modificado, las cuales deben ser de 1. Alternativas (Se devuelve el valor 0 o una excepción): Error: No se han añadido los parámetros necesarios al método. Error: El valor de algún parámetro no es correcto. Error: Fallo técnico en la base de datos. Error: Usuario no encontrado en la base de datos.
Requisito Software Asociado	FSR -03 y FSR -04

211 Prueba Caja Negra 05

PR-CN-05	
Objetivo	Comprobar que un usuario puede loguearse correctamente en la aplicación
Clase	Usuario
Método	LoginUsuario
Entrada	Recibe el nombre y la contraseña del usuario que desea entrar en la aplicación.
Descripción	Se llama al método LoginUsuario pasándole por parámetro el nombre y la contraseña del usuario. A continuación se comprueba en la base de datos el usuario y la contraseña para verificar su acceso.
Salida	Se devuelve true si el usuario es correcto junto con su contraseña. Alternativas (Se devuelve el valor false o una excepción): <ul style="list-style-type: none"> - Error: El nombre de usuario y contraseña no son correctos. - Error: No se han añadido los parámetros necesarios al método. - Error: El valor de algún parámetro no es correcto. - Error: Usuario no encontrado en la base de datos- - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -01

212 Prueba Caja Negra 06

PR-CN-06	
Objetivo	Comprobar que la contraseña de un usuario ha caducado.
Clase	Usuario
Método	ComprobarCaducidad
Entrada	Recibe el nombre del usuario que se desea comprobar
Descripción	Se llama al método ComprobarCaducidad pasándole el nombre de usuario que se quiere comprobar. A continuación se comprueba en la base de datos si la contraseña del usuario ha caducado.
Salida	Se devuelve true si la contraseña ha caducado. Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto. - Error: No se han añadido el parámetro necesario al método. - Error: Usuario no encontrado en la base de datos. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR –03

213 Prueba Caja Negra 07

PR-CN-07	
Objetivo	Comprobar que un usuario ha sido eliminado de la base de datos
Clase	Usuario
Método	EliminarUsuario
Entrada	Recibe el nombre del usuario a eliminar.
Descripción	Se llama al método EliminarUsuario pasándole por parámetro el nombre de usuario a eliminar. A continuación se actualiza la base de datos eliminando el usuario.
Salida	Se devuelve true si el usuario ha sido eliminado. Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto. - Error: No se han añadido el parámetro necesario al método. - Error: Usuario no encontrado en la base de datos. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR –01

214 Prueba Caja Negra 08

PR-CN-08	
Objetivo	Comprobar que un cargamento se abre correctamente
Clase	Cargamento
Método	AbrirCargamento
Entrada	Recibe el nombre del cargamento.
Descripción	Se llama al método AbrirCargamento pasándole por parámetro el nombre del cargamento. A continuación se actualiza la base de datos cambiando el estado del cargamento a abierto.
Salida	Se devuelve true si el cargamento se ha podido abrir. Alternativas (se devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto. - Error: No se han añadido el parámetro necesario al método. - Error: El cargamento pertenece a un envío ya consolidado. - Error: Cargamento no encontrado en la base de datos.
Requisito Software Asociado	FSR -09

215 Prueba Caja Negra 09

PR-CN-09	
Objetivo	Comprobar que un cargamento se cierra correctamente
Clase	Cargamento
Método	CerrarCargamento
Entrada	Recibe el nombre del cargamento.
Descripción	Se llama al método CerrarCargamento pasándole por parámetro el nombre del cargamento. A continuación se actualiza la base de datos cambiando el estado del cargamento a cerrado.
Salida	Se devuelve true si el cargamento se ha podido cerrar. Alternativas:(se devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El contenido del cargamento no incluye ninguna caja. - Error: No se ha encontrado el cargamento en la base de datos. - Error: El contenido del cargamento incluye cajas abiertas. - Error: Fallo en la conexión.
Requisito Software Asociado	FSR -07

216 Prueba Caja Negra 10

PR-CN-10	
Objetivo	Comprobar que un nuevo cargamento se ha creado correctamente.
Clase	Cargamento
Método	CrearCargamento
Entrada	Recibe los datos del nuevo cargamento a crear
Descripción	Se llama al método CrearCargamento pasándole por parámetro los datos del nuevo cargamento. A continuación se actualiza la base de datos añadiendo el nuevo cargamento.
Salida	Se devuelve el número de líneas que se han añadido, las cuales deben ser de 1. Alternativas (devuelve un 0 o una excepción): <ul style="list-style-type: none"> - Error: El valor de algún parámetro no es correcto. - Error: No se han añadido los parámetros obligatorios al método. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -05

217 Prueba Caja Negra 11

PR-CN-11	
Objetivo	Insertar una caja en un cargamento
Clase	Cargamento
Método	InsertarCaja
Entrada	Recibe el cargamento y la caja a insertar
Descripción	Se llama al método InsertarCaja pasándole por parámetro el cargamento y la caja a insertar. A continuación se actualiza la base de datos añadiendo al cargamento la caja insertada.
Salida	Se devuelve true si la caja ha sido insertada correctamente. Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El valor de algún parámetro no es correcto. - Error: No se han añadido los parámetros obligatorios al método. - Error: No se encuentra la caja o el cargamento en la base de datos. - Error: La caja no está cerrada. - Error: El cargamento no está abierto. - Error: El cargamento está lleno. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -20

218 Prueba Caja Negra 12

PR-CN-12	
Objetivo	Crear nuevo envío
Clase	Envío
Método	CrearEnvío
Entrada	Recibe los datos del envío a crear
Descripción	Se llama al método CrearEnvío recibiendo por parámetro los valores del nuevo envío a crear. A continuación se actualiza la base de datos añadiendo el nuevo envío.
Salida	Se devuelve el número de líneas que se han añadido, las cuales deben ser de 1. Alternativas (se devuelve 0 o una excepción): <ul style="list-style-type: none"> - Error: El valor de algún parámetro no es correcto. - Error: No se han añadido los parámetros obligatorios al método. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -40

219 Prueba Caja Negra 13

PR-CN-13	
Objetivo	Obtener envío
Clase	Envío
Método	ObtenerEnvío
Entrada	Recibe el envío a obtener
Descripción	Se llama al método ObtenerEnvío pasándole por parámetro el nombre del envío a obtener. A continuación se consulta en la base de datos el envío y se devuelve toda su información junto con sus cargamentos.
Salida	Datos del envío con el contenido de sus cargamentos Alternativas: <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto - Error: El envío no se encuentra en la base de datos. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -36, FSR -37, FSR -38

220 Prueba Caja Negra 15

PR-CN-15	
Objetivo	Insertar Cargamento en envío
Clase	Envío
Método	InsertarCargamento
Entrada	Recibe el cargamento y el envío donde se quiere insertar
Descripción	Se llama al método InsertarCargamento pasándole por parámetro el cargamento y el envío. A continuación se actualiza la base de datos añadiendo el cargamento al envío.
Salida	<p>Se devuelve true si el cargamento ha sido añadido.</p> <p>Alternativas (devuelve false o una excepción):</p> <ul style="list-style-type: none"> - Error: El valor de algún parámetro no es correcto. - Error: No se han añadido los parámetros obligatorios al método. - Error: No se encuentra el cargamento o el envío en la base de datos. - Error: El cargamento no está cerrado - Error: El envío está lleno. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -24

221 Prueba Caja Negra 16

PR-CN-16	
Objetivo	Abrir caja existente
Clase	Caja
Método	AbrirCaja
Entrada	Recibe la caja que se quiere abrir
Descripción	Se llama al método AbrirCaja pasándole por parámetro el nombre de la caja que se desea abrir. A continuación se actualiza la base de datos cambiando el estado de la caja.
Salida	<p>Se devuelve true si la caja se ha podido abrir.</p> <p>Alternativas (se devuelve false o una excepción):</p> <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto. - Error: No se han añadido el parámetro necesario al método. - Error: La caja pertenece a un cargamento ya consolidado. - Error: Caja no encontrado en la base de datos. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -14

222 Prueba Caja Negra 17

PR-CN-17	
Objetivo	Cerrar caja existente
Clase	Caja
Método	CerrarCaja
Entrada	Recibe la caja que se quiere cerrar
Descripción	Se llama al método CerrarCaja pasándole por parámetro el nombre de la caja. A continuación se actualiza la base de datos cambiando el estado de la caja.
Salida	Se devuelve true si la caja ha sido cerrada correctamente. Alternativas:(se devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El contenido de la caja no incluye ningún pedido. - Error: No se ha encontrado la caja en la base de datos. - Error: Fallo en la conexión
Requisito Software Asociado	FSR -04

223 Prueba Caja Negra 18

PR-CN-18	
Objetivo	Crear nueva caja
Clase	Caja
Método	CrearCaja
Entrada	Recibe los datos de la caja a crear.
Descripción	Se llama al método CrearCaja y se le pasa por parámetro los datos de la nueva caja. A continuación se actualiza la base de datos añadiendo la nueva caja creada.
Salida	Se devuelve el número de líneas que se han añadido, las cuales deben ser de 1. Alternativas (se devuelve 0 o una excepción): <ul style="list-style-type: none"> - Error: El valor de algún parámetro no es correcto. - Error: No se han añadido los parámetros obligatorios al método. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -10

224 Prueba Caja Negra 19

PR-CN-19	
Objetivo	Obtener datos de una caja existente
Clase	Caja
Método	ObtenerDatosCaja
Entrada	Recibe la caja de la cual se quiere obtener la información
Descripción	Se llama al método ObtenerDatosCaja pasándole por parámetro el nombre de la caja. A continuación se consulta la base de datos devolviendo la información de la caja.
Salida	Datos de la caja con el contenido de sus pedidos Alternativas: <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto - Error: La caja no se encuentra en la base de datos. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -37

225 Prueba Caja Negra 20

PR-CN-20	
Objetivo	Insertar Pedido en caja
Clase	Caja
Método	InsertarPedido
Entrada	Recibe la caja y el pedido a insertar en ella.
Descripción	Se llama al método InsertarPedido pasándole por parámetro el pedido y la caja. A continuación se actualiza la base de datos añadiendo dicho pedido a la caja.
Salida	Se devuelve true si el pedido ha sido insertado correctamente. Alternativas (devuelve false o una excepción): Error: La caja no permite la inserción del pedido al sobrepasar los atributos de la caja. <ul style="list-style-type: none"> - Error: La caja no ha sido encontrada en la base de datos. - Error: El pedido no ha sido encontrado en la base de datos. - Error: Fallo en la conexión.
Requisito Software Asociado	FSR -15

226 Prueba Caja Negra 21

PR-CN-21	
Objetivo	Obtener datos de un pedido recepcionado
Clase	Pedido
Método	ObtenerDatosPedido
Entrada	Recibe el pedido del que se desea la información.
Descripción	Se llama al método ObtenerDatosPedido pasándole por parámetro el nombre del pedido que se desea. A continuación se consulta la base de datos devolviendo la información de dicho pedido.
Salida	Datos del pedido Alternativas: <ul style="list-style-type: none"> - Error: El valor del parámetro no es correcto - Error: El pedido no se encuentra en la base de datos. - Error: Fallo técnico en la base de datos
Requisito Software Asociado	FSR -36, FSR -37, FSR -38

227 Prueba Caja Negra 22

PR-CN-22	
Objetivo	Dividir un pedido para su posterior consolidación
Clase	Pedido
Método	DividirPedido
Entrada	Recibe el pedido del que se desea particionar y el número de partes lógicas.
Descripción	Se llama al método DividirPedido pasándole por parámetro el pedido a particionar junto con sus partes lógicas. A continuación se actualiza la base de datos particionando el pedido.
Salida	Se devuelve true si el pedido ha sido particionado Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El pedido no existe en la base de datos. - Error: Los parámetro son incorrectos. - Error: Ausencia de algún parámetro en el método. - Error: Fallo en la conexión.
Requisito Software Asociado	FSR -15

228 Prueba Caja Negra 23

PR-CN-23	
Objetivo	Recepcionar un pedido en el almacén.
Clase	Pedido
Método	RecepcionarPedido, RecepcionarReparable, RecepcionarPedidoCliente
Entrada	Recibe el identificador del pedido y actualiza la base de datos.
Descripción	Para probarlo se llama por separado a cada uno de los métodos RecepcionarPedido, RecepcionarReparable, RecepcionarPedidoCliente y se le pasa por parámetro el identificador del pedido.
Salida	Se devuelve true si el pedido ha sido recepcionado correctamente. Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El pedido no se encuentra en la base de datos. - Error: El parámetro está vacío. - Error: El valor del parámetro es incorrecto. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -29, FSR -30, FSR -31

229 Prueba Caja Negra 24

PR-CN-24	
Objetivo	Consolidar y posibilitar un envío.
Clase	Pedido
Método	ConsolidarPedido, ConsolidarReparable, EnvioClienteFin
Entrada	Recibe el identificador del pedido y actualiza la base de datos.
Descripción	Para probarlo se llama a cada uno de los métodos ConsolidarPedido, ConsolidarReparable, EnvioClienteFin pasándole el identificador del pedido a consolidar. A continuación se actualiza la base de datos cambiando el estado del pedido.
Salida	Se devuelve true si el pedido ha sido consolidado correctamente. Alternativas (devuelve false o una excepción): <ul style="list-style-type: none"> - Error: El pedido no se encuentra en la base de datos. - Error: El parámetro está vacío. - Error: El valor del parámetro es incorrecto. - Error: Cargamentos abiertos dentro del pedido - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -31, FSR -32, FSR -33, FSR -34

230 Prueba Caja Negra 25

PR-CN-25	
Objetivo	Obtener información de un pedido.
Clase	Pedido
Método	ObtenerPedidosEnviados,ObtenerPedidosNoProcesados,ObtenerPedidosPendientes
Entrada	No tiene
Descripción	Para probar se llama a cada uno de los métodos ObtenerPedidosEnviados,ObtenerPedidosNoProcesados y ObtenerPedidosPendientes. A continuación se consulta en la base de datos dichos pedidos devolviendo la información de cada uno.
Salida	Información del pedido. Alternativas (devuelve null o excepción): - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -36, FSR -37, FSR -38

231 Prueba Caja Negra 26

PR-CN-26	
Objetivo	Añadir una incidencia a un pedido
Clase	Pedido
Método	AltaIncidencia
Entrada	Pedido con los datos de su incidencia
Descripción	Se llama al método AltaIncidencia pasándole por parámetro el pedido junto con sus datos de incidencia. A continuación se actualiza el pedido en la base de datos para añadir su incidencia.
Salida	Se devuelve true si la incidencia ha sido dada de alta correctamente. Alternativas (devuelve false o una excepción): - Error: El pedido no existe en la base de datos - Error: Valores de los parámetros no válidos. - Error: Ausencia de campos obligatorios en los parámetros. - Error: Fallo técnico en la base de datos.
Requisito Software Asociado	FSR -42

9.2 Especificación De Pruebas De Integración

El objetivo es comprobar que la comunicación con cada uno de los sistemas externos funcione correctamente cuando se integran al proyecto.

La estrategia a seguir para comprobar que el sistema de central de almacén se comunica con los demás sistemas, es la estrategia de integración incremental de arriba abajo y de abajo arriba. Se lanzarán comunicaciones contra cada uno de los sistemas externos y se verificará que ambos sistemas envían y reciben datos. Posteriormente se van lanzando comunicaciones con cada uno de los sistemas según se vayan integrando estos al sistema principal.

Los criterios de aceptación se basan en que la información recibida de los sistemas externos siga el formato que se espera recibir y por lo tanto sea reconocido por el mismo, además que la información enviada no es rechazada por tener un formato no valido.

Para saber que existe una comunicación correcta, se deberán reflejar todas las líneas de estándar FMS en las bases de datos:

- Generación de la línea FMS XDI a la entrada del pedido en el almacén de EEUU.
- Generación de la línea FMS XDS a la salida del pedido del almacén de EEUU.
- Generación de la línea FMS XDL de notificación de llegada del pedido al almacén del cliente.
- Generación de la línea FMS a la salida del pedido del almacén del cliente para su entrega final.
- Generación de la línea FMS XDR a la entrada del pedido reparable en el almacén de EEUU.
- Generación de la línea FMS XDT a la salida del pedido reparable desde el almacén del cliente.

9.3 Especificación De Pruebas De Inserción De Datos

Se deberá comprobar que el script de creación e inserción de tablas y datos junto con sus procedimientos almacenados es correcto.

Para ello el script deberá ejecutarse sin ningún problema obteniendo el mensaje por pantalla "Script ejecutado correctamente".

Los posibles errores que pueden originarse son:

- Versión de Sql Server no compatible. (Se recuerda que la aplicación se desarrolla en la versión de Sql Server 2008.

- Script incompleto. No se ha podido cargar completamente el script en la herramienta de Sql Server.
- Servidor no disponible.

10 Implantación y explotación

10.1 Instalación

10.1.1 Cliente

Antes de comenzar con la realización de la instalación del sistema al entorno de operación a nivel de software, es necesario realizar una verificación de la correcta instalación y funcionamiento de los diversos equipos que se emplearán para el uso y mantenimiento del sistema.

El primer paso a realizar antes de comenzar con la instalación de la infraestructura del sistema, será conocer los requisitos de los productos que vamos a instalar. Una vez se verifique que estos requisitos se cumplen, se comprobará que el software necesario para instalar se encuentra a plena disposición en las versiones necesarias.

Describiremos ahora las tareas necesarias para la instalación de nuestra aplicación. Comenzando por el sistema operativo sobre el que se ejecutará, continuando por el gestor de la base de datos, el servidor web, los entornos necesarios y terminando con el despliegue de nuestra aplicación.

Instalación del sistema operativo **Windows XP**:

- Realización de las actualizaciones necesarias del sistema.
- Realización de la configuración necesaria de las opciones de red.
- Realización de las actualizaciones propias del explorador utilizado por defecto.

10.1.2 Servidor

10.1.2.1 Instalación del internet information server

- Configuración del pool de conexiones desde el servidor de aplicaciones a la base de datos.
- Configuración de los puertos disponibles para nuestra aplicación.
- Configuración de la seguridad del servidor web.

10.1.2.2 Publicación del sitio web

Despliegue de nuestra aplicación en el servidor web:

- Verificación del correcto despliegue mediante logs y mensajes del propio servidor web de Windows junto con la comprobación de la tecnología ASP.Net.

10.1.2.3 Instalación del gestor de bases de datos Sql Server 2008:

- Creación de la base de datos.
- Creación del usuario que administrará dicha base de datos.
- Ejecución del script de creación de las tablas necesarias para la base de datos, según lo establecido en el apartado *Diseño Físico de Datos* de dicho documento.

10.1.2.4 Establecimiento de permisos

Se deben configurar los permisos designados por el administrador para poder administrar el sitio web desde la consola de IIS visualizando su propia interfaz del servidor en caso de error o fallo del propio sistema.

11 Gestión del Proyecto

11.1 Introducción

Como se observa en la siguiente figura , existen varios factores que determinan la calidad del software y la eficiencia de la organización, entre ellos están la complejidad del producto, las tecnologías y las personas, así como algunas condiciones de entorno que también tienen su impacto, estas pueden ser condiciones de gestión (Ej.: plazo de entrega, regla de empresa), entornos de desarrollo y características del cliente, sin embargo en el centro de todas ellas se encuentra el proceso pues es el único factor de los controlables al mejorar la calidad del software y su rendimiento como organización.

Analizando y mejorando el proceso se puede obtener mejores productos.



50 Calidad Del Software

11.2 Metodología y ciclo de vida

La metodología seguida es la llama Métrica 3. Se han usado los puntos más representativos para poder realizar dicho proyecto, de tal forma que se ha realizado un análisis exhaustivo incluyendo el estudio del estándar FMS mediante una lógica de procesos y requisitos necesarios para el desarrollo. Se ha incluido un diseño muy representativo junto con el desarrollo de las pruebas unitarias para obtener una aplicación gestor de almacén bajo el estándar adoptado. Así se obtiene una aplicación y una documentación necesaria para el entendimiento del mismo de gran calidad.

En cuanto a la selección del ciclo de vida, primero se detalla la información que ha sido seguida para seleccionar el ciclo de vida que más se adaptara a nuestro proceso de desarrollo.

El ciclo de vida consiste en determinar:

- Las fases productivas de un proyecto.
- Los objetivos de cada fase productiva, y
- Los productos obtenidos en cada una de estas fases así como sus características

Se han propuesto muchos ciclos de vida para el desarrollo del software, pero estos son los más representativos que se pueden adaptar a nuestro desarrollo:

- Ciclo de vida clásico o en cascada.
- Construcción rápida de Prototipos Desechables
- Incremental
- Evolución de prototipos
- Reutilización de Software
- Síntesis automática de software
- En espiral.
- Ciclo de vida en “V”.

El elegido es el ciclo de vida en “V”.

Modelo en V:

La unión mediante líneas discontinuas entre las fases de la parte izquierda y las pruebas de la derecha representa una doble información. Por un lado sirve para indicar en qué fase de desarrollo se deben definir las pruebas correspondientes. Por otro sirve para saber a qué fase de desarrollo hay que volver si se encuentran fallos en las pruebas correspondientes.

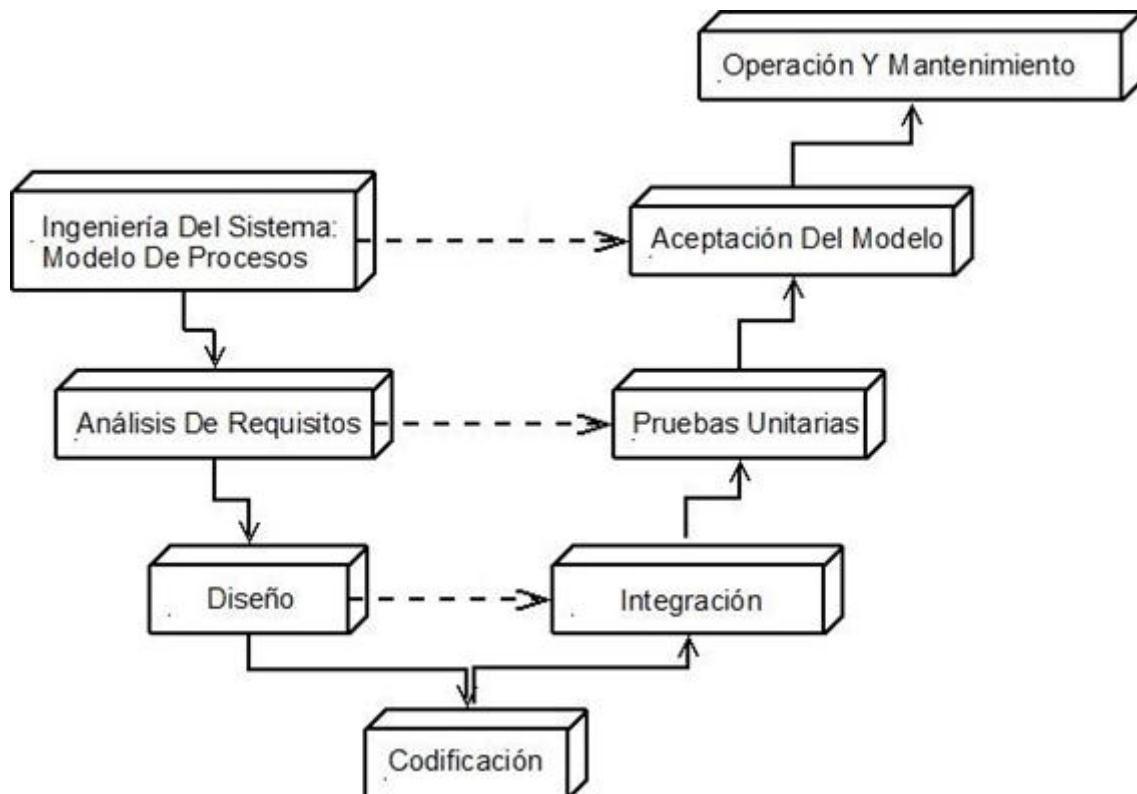
Por lo tanto el modelo en V hace más explícita parte de las iteraciones y repeticiones de trabajo que están ocultas en el modelo en cascada. Mientras el foco del modelo en cascada se sitúa en los documentos y productos desarrollados, el modelo en V se centra en las actividades y la corrección.

Ventajas:

- La relación entre las etapas de desarrollo y los distintos tipos de pruebas facilitan la localización de fallos
- Es un modelo sencillo y de fácil aprendizaje
- Hace explícito parte de la iteración y trabajo que hay que revisar
- Especifica bien los roles de los distintos tipos de pruebas a realizar

Éste es el ciclo de vida que más se asemeja al modo de trabajo que ha sido empleado en dicho proyecto.

El ciclo de vida desarrollado es el siguiente:



51 Ciclo De Vida

11.3 Estimación de recursos temporales

El proyecto deberá ser finalizado el día 25 de Mayo de 2012 y entregado antes del día 15 de Junio de 2012.

El comienzo se desarrolló a mediados de Diciembre de 2011 optimizando la gestión de tiempo para poder sobrellevar posibles riesgos de retrasos en las entregas.

11.4 Estimación de recursos económicos

11.4.1 Recursos materiales

- Ordenador (2nd generation Intel® Core™ i3-2370M con procesador de 2.40 Ghz, y 3M cache con DDR3 SDRAM de 4.096 MB a 1.333 MHz canal dual), incluyendo todos los periféricos como ratón, teclado, lector de DVD y sistema operativo Windows 7. 1500 €
- Impresora Láser para imprimir documentación necesaria del proyecto. 150 €

11.4.2 Recursos humanos

A continuación se muestra la relación de participantes en el proyecto junto con su sueldo por hora.

Documentalista: 20 €/h

Analista Funcional: 55€/h

Jefe De Proyecto: 100 €/h

Arquitecto De Sistemas: 55€/h

Desarrollador: 30€/h

Técnico: 25 €/h

11.4.3 Costes totales

Sumando los costes materiales a los costes de recursos humanos obtenemos:

$1350\text{€} + 43985\text{€} = 45.335\text{ €}$ Sin IVA aplicado

52.520€ con IVA Aplicado (16%)

A continuación en la siguiente página se muestra el desglose del coste de cada tarea ordenados de mayor a menos coste.

52 Costes De Las Tareas

Id	Nombre de tarea	Costo total
38	Definición de la lógica de procesos	4.400,00 €
35	Elaboración del modelo de datos	3.520,00 €
36	Elaboración del Modelo Lógico de Datos	3.520,00 €
48	Descripción de Métodos de las Operaciones	3.080,00 €
10	Especificación de requisitos	2.640,00 €
43	Diseño de las clases	1.760,00 €
47	Identificación de operaciones de negocio	1.320,00 €
62	Gestión De Cajas y Cargamentos	1.200,00 €
55	Pruebas caja negra	1.045,00 €
9	Estado del Arte / Estado de la cuestión	960,00 €
63	Proceso Consolidación	960,00 €
67	Gestión De Cajas	960,00 €
31	Análisis de los casos de uso	880,00 €
54	Pruebas caja blanca	880,00 €
56	Pruebas de inserción de datos	880,00 €
57	Pruebas de integración	880,00 €
61	Consulta Pedidos	720,00 €
64	Gestión Incidencias	720,00 €
66	Gestión de Usuarios	720,00 €
68	Gestión De Cargamentos	720,00 €
69	Consultas FMS	720,00 €
72	Autenticación Usuario	720,00 €
74	Consultas FMS	720,00 €
60	Autenticación Usuario	480,00 €
70	Gestión de Incidencias	480,00 €
73	Gestión de Cajas	480,00 €
75	Gestión de Cargamentos	480,00 €
30	Especificación de Interfaces con otros sistemas	440,00 €
32	Especificación de Casos de Uso	440,00 €
33	Identificación de Clases Asociadas	440,00 €
34	Descripción de la Interacción de Clases	440,00 €
40	Definición de Niveles de Arquitectura	440,00 €
45	Diseño de Asociaciones y agregados	440,00 €
46	Identificación de atributos de las clases	440,00 €
39	Definición de la arquitectura del sistema	220,00 €
44	Identificación de clases adicionales	220,00 €
51	Diseño del Modelo Físico de Datos	220,00 €
13	Estimación de recursos temporal	200,00 €
14	Estimación de recursos económicos	200,00 €
17	Costes totales	200,00 €
77	Instalación	200,00 €
80	Conclusiones	200,00 €
81	Trabajos futuros	200,00 €
8	Estructura del documento	160,00 €
82	Referencias	160,00 €
83	Anexo A. Glosario	160,00 €
41	Especificación del Entorno Tecnológico	110,00 €
15	Recursos materiales	100,00 €
16	Recursos humanos	100,00 €
19	Fases del proyecto	100,00 €
4	Introducción a la logística militar	80,00 €
24	Microsoft Visual Studio 2010	60,00 €
29	Comparativa Sql Server y Mysql.	60,00 €
42	Especificación de Requisitos de Calidad	55,00 €
52	Especificación de la Distribución de Datos	55,00 €
5	Introducción a Foreign Military Service	40,00 €
6	Motivación.	40,00 €
7	Objetivos	40,00 €
22	Microsoft Windows 7	40,00 €
23	Integración con Internet Information Services	40,00 €
79	Aspectos legales	40,00 €
25	Microsoft Office 2010	20,00 €
26	Microsoft Visio 2010	20,00 €
27	Aris Express	20,00 €
28	SQL Server 2008	20,00 €
		43.985,00 €

11.5 Gestión Del Riesgo

Para analizar los riesgos utilizaremos tablas con el siguiente formato:

ID de riesgo:	Prob:	Impacto:	
Descripción:			
Refinamiento/contexto:			
Mitigación/supervisión:			
Gestión/plan de contingencia/disparador:			

ID de riesgo será el código que identifique cada riesgo.

Fecha la fecha en la que se recogió en el plan de gestión de la calidad.

Prob (Probabilidad) la probabilidad (en %) de que se dé el riesgo.

Impacto medirá el daño que puede provocar el riesgo al proyecto (de 'muy alto' a 'muy bajo').

Descripción será la descripción del riesgo.

Refinamiento o Contexto aportará detalles sobre la descripción o producción del riesgo.

Mitigación/supervisión hace mención a las directrices que se llevarán a cabo para manejar y atajar el riesgo.

Gestión/plan de contingencia/disparador recogerá la actuación o plan de contingencia para paliar los efectos ocasionados por el percance así como el suceso que lo ocasionaría.

A continuación, presentamos una relación de los riesgos que pueden darse en el desarrollo del proyecto.

11.5.1 Riesgos Posibles

232 Riesgo Cliente No Disponible

ID de riesgo: Cliente No Disponible	Prob: 10%	Impacto: alto	
Descripción: El cliente no está disponible para la revisión de los documentos y/o para tener su validación o corrección del proyecto.			
Refinamiento/contexto: <i>Subcondición 1:</i> El cliente no puede tener reuniones debido a su apretada agenda. <i>Subcondición 2:</i> La implicación del cliente es baja y por lo tanto no propone o acepta reuniones.			
Mitigación/supervisión: Minimizar el tiempo con el cliente al máximo. Hay que hablar con los stakeholders para evitar la presión sobre el cliente.			
Gestión/plan de contingencia/disparador: Asignar en el calendario las reuniones con el cliente para buscar los huecos posibles para las validaciones. <i>Disparador:</i> No se asignan fechas para validaciones del documento.			

233 Riesgo Tareas Faltantes, Poco Esfuerzo, Retraso De Entregas

ID de riesgo: Tareas Faltantes	Prob: 40%	Impacto: medio	
Descripción: Tareas o Hitos faltantes en los calendarios como revisiones o reuniones de revisión.			
Refinamiento/contexto: <i>Subcondición 1:</i> La falta de gestión de tareas puede conllevar a retrasos en el proyecto. <i>Subcondición 2:</i> La falta de hitos de tareas puede conllevar a retrasos en el proyecto.			
Mitigación/supervisión: Tener una buena gestión de tareas. Tener una buena planificación previa y durante el proyecto de los hitos.			
Gestión/plan de contingencia/disparador: Plantear fechas de emergencia y tener un margen para posibles fechas conflictivas <i>Disparador:</i> Se retrasan las fechas de entrega debido a la mala planificación. Se lleva una gestión del tiempo de tal forma que si hay retrasos se puedan sobreponer.			
ID de riesgo: Poco Esfuerzo	Prob: 10%	Impacto: medio	
Descripción: Se ha establecido un calendario demasiado relajado para el desarrollo del proyecto			
Refinamiento/contexto: <i>Subcondición 1:</i> Los alumnos tienen tiempo extra. <i>Subcondición 2:</i> Se han planteado fechas de revisión demasiado relajadas que conllevan falta de tiempo de trabajo efectivo.			
Mitigación/supervisión: Revisar periódicamente los calendarios para ajustar el tiempo de trabajo. Revisar históricamente el trabajo efectivo.			
Gestión/plan de contingencia/disparador: Plantear fechas de emergencia y tener un margen para posibles fechas conflictivas <i>Disparador:</i> Se retrasan las fechas de entrega debido a la mala planificación.			
ID de riesgo: Retraso de entregas	Prob: 20%	Impacto: alto	
Descripción: Las entregas establecidas se retrasan.			
Refinamiento/contexto: <i>Subcondición 1:</i> Se retrasan las entregas/revisiones del proyecto por falta de tiempo. <i>Subcondición 2:</i> Se detecta falta de tiempo en las reuniones de seguimiento.			
Mitigación/supervisión: Revisar las fechas periódicamente para ver que no se están superando las estimadas.			
Gestión/plan de contingencia/disparador: Se retrasa la fecha de la entrega en la que se ha detectado falta de tiempo. <i>Disparador:</i> Se detecta falta de tiempo para las entregas.			

234 Riesgo Pérdida De Personas, Desmotivación, Falta De Decisiones, Desastre Natural

ID de riesgo: Pérdida de personas	Prob: 20%	Impacto: muy alto	
Descripción: Pérdida de personas clave.			
Refinamiento/contexto: Subcondición 1: La pérdida o abandono de algún miembro del equipo ocasionaría problemas en el desarrollo del proyecto. Subcondición 2: Dependiendo de la importancia del miembro que deje el proyecto, podrían peligrar continuidad o éxito.			
Mitigación/supervisión: Tener una buena gestión del proyecto en todo momento para que no pueda ocurrir dicha pérdida o abandono.			
Gestión/plan de contingencia/disparador: Entablar negociaciones con el miembro que plantee su marcha del proyecto. Disparador: Baja implicación por parte de algún miembro o aviso formal de abandono.			
ID de riesgo: Desmotivación	Prob: 5%	Impacto: medio	
Descripción: Baja moral en el equipo, desmotivación.			
Refinamiento/contexto: Subcondición: Una desmotivación general provoca ralentización en todo el proceso además de fallos u errores generados por distracción.			
Mitigación/supervisión: Tener una buena planificación y gestión por parte de los integrantes del proyecto para mantener la cohesión del equipo y la motivación de los miembros.			
Gestión/plan de contingencia/disparador: Iniciar labores de trabajo en equipo y coaching con los integrantes del equipo. Disparador: Baja implicación por parte de algún miembro.			
ID de riesgo: Falta de decisiones	Prob: 40%	Impacto: medio	
Descripción: Demoras en una toma de decisiones en la organización			
Refinamiento/contexto: Subcondición 1: Las decisiones importantes deben ser estudiadas conjuntamente y la toma debe ser clara y concisa, ya que muchos procesos pueden estar a la espera de esa decisión tomada.			
Mitigación/supervisión: Buena gestión y planificación de tareas por parte de los integrantes del proyecto.			

Gestión/plan de contingencia/disparador: Plantear fechas de emergencia y tener un margen para posibles fechas conflictivas <i>Disparador: Se retrasa la validación y aprobación de alguna actividad. Sobrecarga de trabajo de los responsables.</i>			
ID de riesgo: Desastre natural	Fecha: 05/10/11	Prob: 10%	Impacto: alto
Descripción: Desastres naturales y problemas climáticos.			
Refinamiento/contexto: <i>Subcondición 1:</i> Cualquier desastre natural o problema desmesurado que pueda darse, entre los que se pueden incluir, por ejemplo, tornados o incendios.			
Mitigación/supervisión: Uso de sistemas de <i>backup</i> para mantener a salvo los datos. Instalación sistemas de defensa ante desastres naturales o incendios.			
Gestión/plan de contingencia/disparador: Plantear protocolo de emergencia y de recuperación. <i>Disparador: Acaecimiento de algún desastre en las instalaciones.</i>			

11.6 Plan del proyecto

11.6.1 Fases del proyecto

Se han establecido las siguientes fases o núcleos:

- Documentación inicial.
- Análisis del sistema.
- Diseño del sistema.
- Desarrollo de pruebas.
- Desarrollo.
- Implantación y explotación.
- Documentación final.

Detallando cada una de ellas obtenemos:

Documentación inicial:

235 Tareas Documentación Inicial

Nombre de tarea
Introducción
Introducción a la logística militar
Introducción a Foreign Military Sales (FMS)
Motivación.
Objetivos
Estructura del documento
Estado del Arte / Estado de la cuestión
Especificación de requisitos
Gestión del Proyecto
Metodología
Estimación de recursos temporales
Estimación de recursos económicos
Recursos materiales
Recursos humanos
Costes totales
Plan del proyecto
Fases del proyecto

Análisis del sistema:

236 Tareas Análisis Del Sistema

Nombre de tarea
Herramientas
Microsoft Windows 7
Integración con Internet Information Services
Microsoft Visual Studio 2010
Microsoft Office 2010
Microsoft Visio 2010
Aris Express
SQL Server 2008
Comparativa Sql Server y Mysql.
Especificación de Interfaces con otros Sistemas
Análisis de los casos de uso
Especificación de Casos de Uso
Identificación de Clases Asociadas a un Caso de Uso
Descripción de la Interacción de Objetos
Elaboración del modelo de datos
Elaboración del Modelo Lógico de Datos

Diseño del sistema:

237 Tareas Diseño Del Sistema

Nombre de tarea
Definición de la lógica de procesos
Definición de la arquitectura del sistema
Definición de Niveles de Arquitectura
Especificación del Entorno Tecnológico
Especificación de Requisitos de Operación y Seguridad
Diseño de las clases
Identificación de clases adicionales
Diseño de Asociaciones y agregaciones
Identificación de atributos de las clases
Identificación de operaciones de las clases
Descripción de Métodos de las Operaciones
Diseño de la arquitectura de módulos del sistema
Diseño físico de datos
Diseño del Modelo Físico de Datos
Especificación de la Distribución de Datos

Desarrollo de pruebas:

238 Tareas Desarrollo De Pruebas

Nombre de tarea
Pruebas caja blanca
Pruebas caja negra
Pruebas de inserción de datos
Pruebas de integración

Desarrollo:

239 Tareas Desarrollo

Nombre de tarea
Interfaz de usuario
Autenticación Usuario
Consulta Pedidos
Gestión De Cajas y Cargamentos
Proceso Consolidación
Gestión Incidencias
Lógica de negocio
Gestión de Usuarios
Gestión De Cajas
Gestión De Cargamentos
Consultas FMS
Gestión de Incidencias
Acceso a datos
Autenticación Usuario
Gestión de Cajas
Consultas FMS
Gestión de Cargamentos

Implantación y explotación.

240 Tareas Implantación Y Explotación

Nombre de tarea
Instalación

Documentación final.

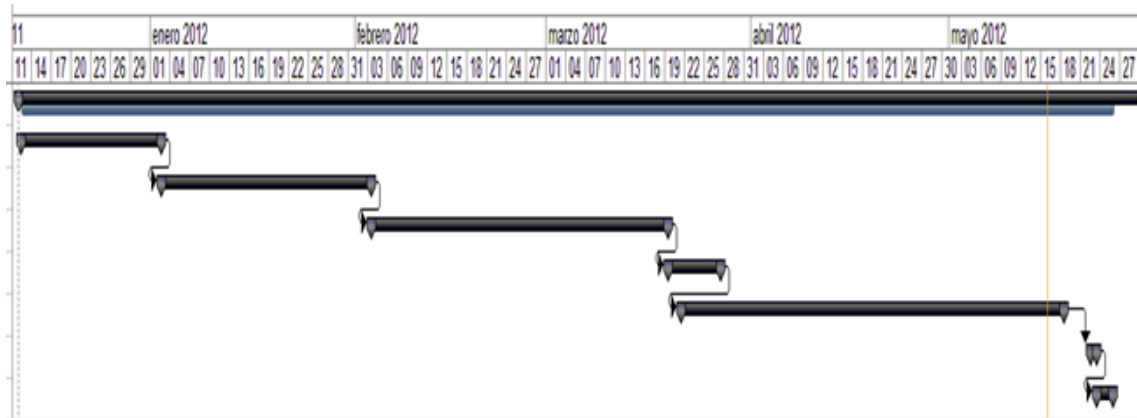
241 Tareas Documentación Final

Nombre de tarea
Aspectos legales
Conclusiones
Trabajos futuros
Referencias
Anexo A. Glosario

11.6.2 Diagrama De Gantt

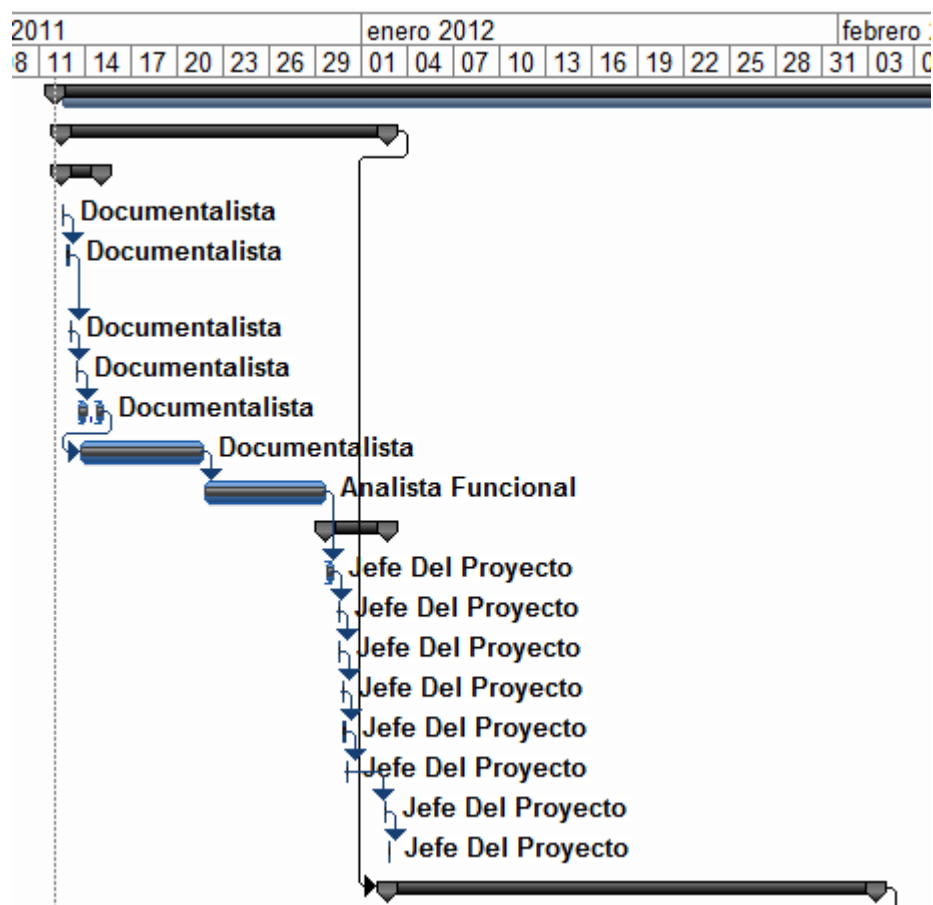
Diagrama General:

53 Diagrama De Gantt General



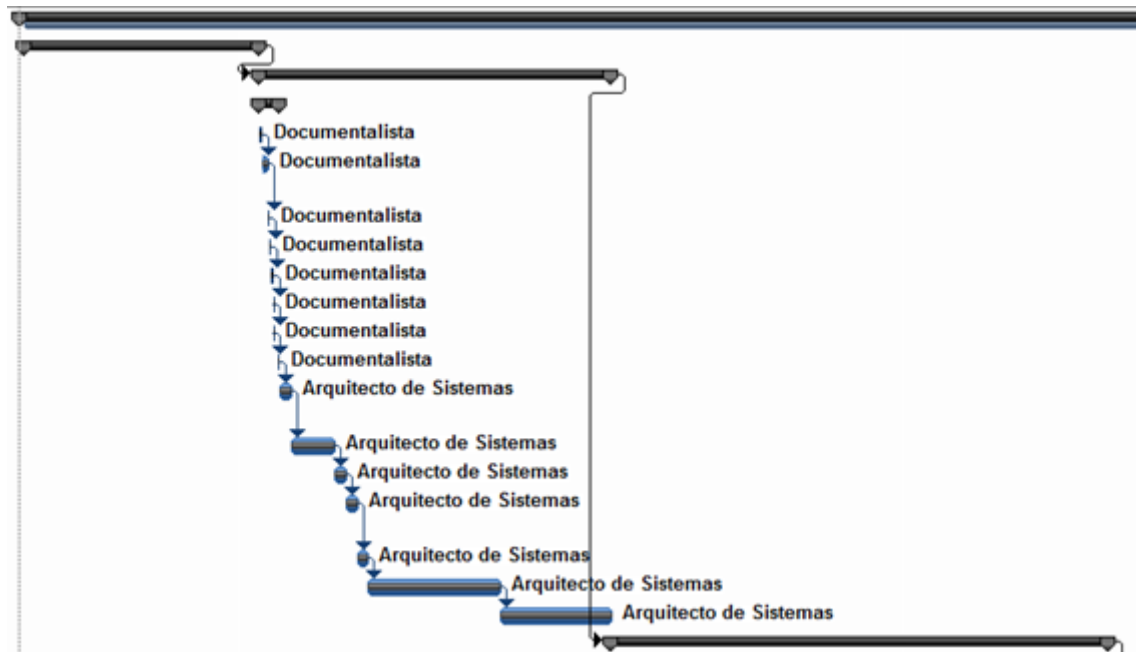
Tareas Documentación Inicial

54 Diagrama De Gantt Documetación Inicial



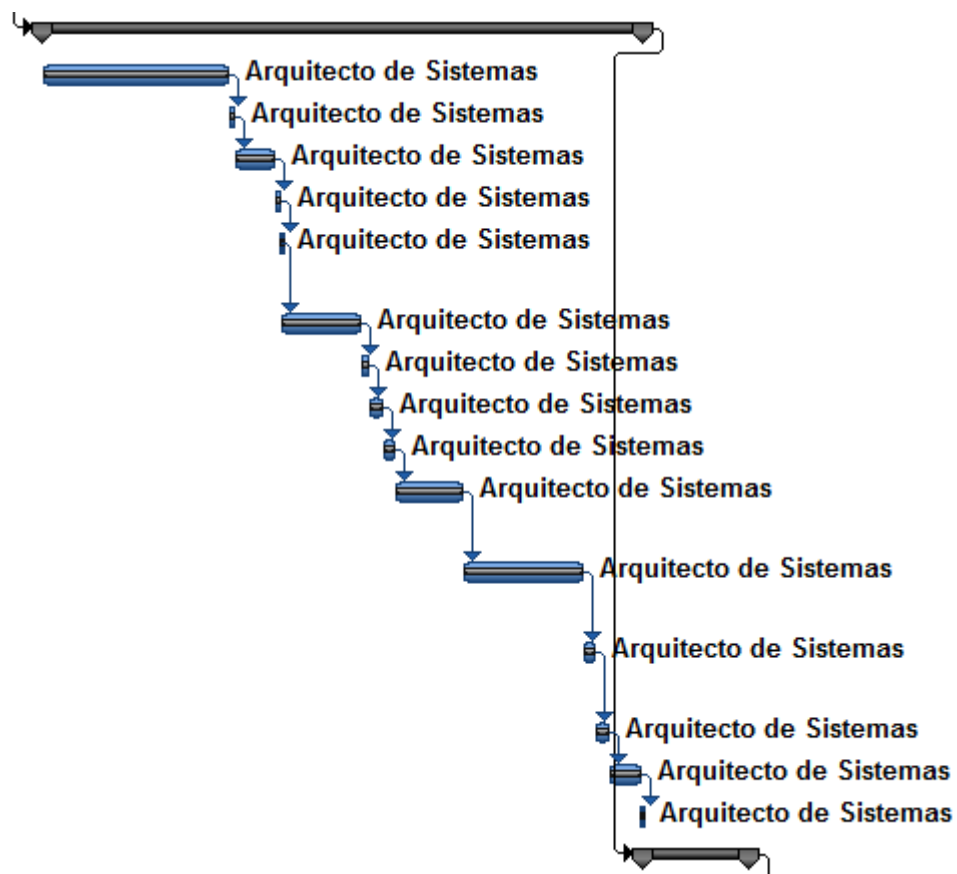
Tareas Análisis Del Sistema

55 Diagrama De Gantt Análisis Del Sistema



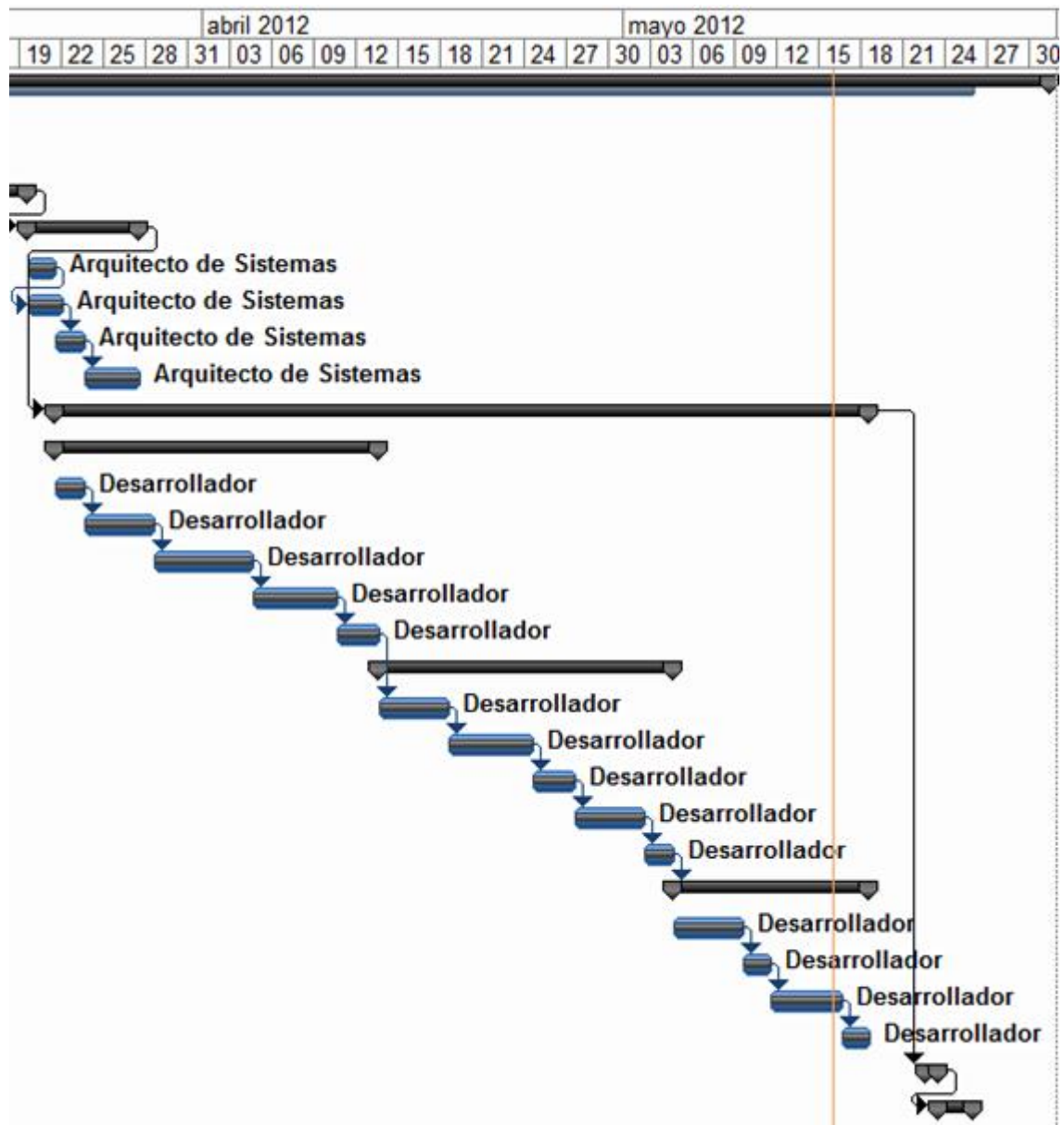
Tareas Diseño Del Sistema

56 Diagrama De Gantt Diseño Del Sistema



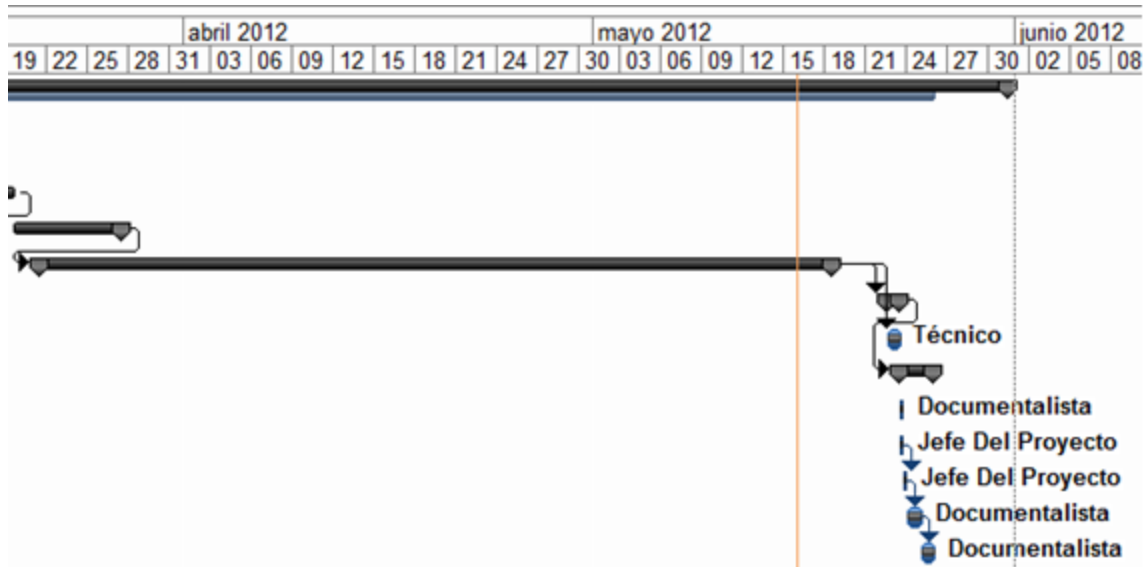
Tareas del Desarrollo Y Pruebas Y Desarrollo

57 Diagrama De Gantt Desarrollo Y Pruebas



Implantación y Explotación y Documentación Final

58 Gantt Documentación Final



11.7 Herramientas

11.7.1 Microsoft Windows 7



Windows 7 es la versión más reciente de Microsoft Windows, línea de sistemas operativos producida por Microsoft Corporation.

A diferencia del gran salto arquitectónico y de características que sufrió su antecesor Windows Vista con respecto a Windows XP, Windows 7 fue concebido como una actualización incremental y focalizada de Vista y su núcleo NT 6.0, lo que permitió mantener cierto grado de compatibilidad con aplicaciones y hardware en los que éste ya era compatible.

Las versiones cliente de Windows 7 fueron lanzadas en versiones para arquitectura 32 bits y 64 bits en las ediciones Home Basic, Home Premium, Professional y Ultimate. No obstante, las versiones servidor de este producto fueron lanzadas exclusivamente para arquitectura 64 bits.

Esto significa que las versiones cliente de 32 bits aún soportan programas Windows 16 bits y MS-DOS. Y las versiones 64 bits (incluyendo todas las versiones de servidor) soportan tanto programas de 32 como de 64 bits.

La aplicación al estar desarrollada en Visual Studio .Net 2010, el sistema operativo bajo el que puede ejecutarse es Windows con Framework 4.0. Por ésta razón, entre otras, desechamos otros sistemas operativos basados Unix o Mac.

242 Requisitos Mínimos Windows 7

Requisitos de hardware mínimos recomendados para Windows 7		
Arquitectura	32 bits	64 bits
Procesador	1 GHz	
Memoria RAM	1 GB de RAM	2 GB de RAM
Tarjeta gráfica	Dispositivo de gráficos DirectX 9 con soporte de controladores WDDM 1.0 (para Windows Aero)	
Disco duro	16 GB de espacio libre	20 GB de espacio libre
Unidad óptica	DVD-R	

11.7.2 Integración con Internet Information Services



IIS (Internet Information Services), es una serie de servicios para los sistemas basados en Windows.

Convierte a un sistema en un servidor de Internet o Intranet de tal manera que los equipos que tienen este servicio instalado pueden publicar páginas como remota.

IIS ofrece servicios de seguridad y métodos de autenticación y las últimas tecnologías de cifrado y autenticación mediante certificados de cliente y servidor.

Una de las formas que tiene de asegurar los datos es mediante SSL, por lo que se proporciona un método para transferir datos entre cliente y servidor de forma segura.

La versión a utilizar es la IIS 7, compatible con Windows 7.

Las características más importantes de IIS son:

- Dirección centralizada de la web.
- Soporte de migración.
- Sincronización de paquetes y archivos.
- Escalabilidad.
- Seguridad, protección del servidor y acceso
- Herramientas de diagnóstico y test.
- Potentes herramientas de administración.
- Acceso con control remoto.

11.7.3 Microsoft Visual Studio 2010



El entorno de desarrollo elegido es *Microsoft Visual Studio* propio para el sistema operativo Windows.

Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

La versión que es utilizada para el desarrollo e implementación es la *2010 Professional*.

Uno de los mayores logros de la versión 2010 de Visual Studio ha sido el de incluir las herramientas para desarrollo de aplicaciones para Windows 7,

tales como herramientas para el desarrollo de las características de Windows 7 (System.Windows.Shell).

Entre sus más destacables características, incluye capacidades para pruebas de unidades dentro del entorno de desarrollo integrado que pueden generar todos los componentes de métodos necesarios, lo que ayuda a garantizar que cada unidad de código funcione correctamente.

Además de esto, aparece una edición que compila las características de todas las ediciones comunes de Visual Studio: Professional, Team Studio, Test, conocida como Visual Studio Ultimate.

Una característica encontrada a nivel personal es la personalización del área de trabajo. Dicha característica ayuda a organizar el entorno digital mediante la compatibilidad con varios monitores, lo que facilita la administración del modo de trabajo y por consiguiente resulta más agradable en el desarrollo y pruebas de la aplicación.

Un punto en contra es que aunque los requisitos de Hardware no sean muy altos, para el uso óptimo de entorno es necesario cómo mínimo 2 gigas de Memoria Ram y un procesador de doble núcleo

243 Requisitos Software Mínimos Visual Studio 2010

Requisitos de software mínimos recomendados para Visual Studio 2010	
Windows XP (x86)	Service Pack 3, excepto Starter Edition
Windows Vista (x86 y x64)	Service Pack 1, excepto Starter Edition
Windows 7	(x86 y x64)
Windows Server 2003 (x86 y x64)	con Service Pack 2
Windows Server 2003 R2 (x86 y x64)	---
Windows Server 2008 (x86 y x64)	con Service Pack 2
Windows Server 2008 R2 (x64)	---

Requisitos de hardware mínimos recomendados para Visual Studio 2010		
Arquitectura	32 bits	64 bits
Procesador	1,6 GHz	
Memoria RAM	1 GB de RAM	
Tarjeta gráfica	Dispositivo de gráficos DirectX 9 con soporte de controladores WDDM 1.0 (para Windows Aero)	
Disco duro	3 GB de espacio libre	
Unidad óptica	DVD-R	

11.7.3.1 Alternativas:

En caso de haber elegido otro lenguaje de programación, hubiéramos utilizado otra plataforma de desarrollo. A continuación se muestra una comparativa en caso de haber utilizado Java2E con el uso de Eclipse Enterprise en vez de ASP.Net con Microsoft Visual Studio.



Ambas interfaces son similares, con varios cuadros centrándose en los proyectos creados (abiertos o cerrados) y las clases u objetos de nuestros proyectos.

En ambos casos, cuando se quiere lanzar la aplicación web, se tiene que tener un servidor ejecutándose en nuestra máquina.

En el caso de Microsoft Visual Studio se crea un servidor propio de aplicaciones en localhost por un puerto automáticamente sin tener que configurar dicho puerto o servidor, por lo que ejecutar una aplicación web es muy sencillo. Además es posible usar otro servidor, por ejemplo IIS.

Por otro lado, Eclipse no posee un servidor propio para poder aplicarse a la propia máquina. Para poder ejecutarse, es necesario instalar un servidor desde Eclipse añadiéndolo en su entorno. En vez de poder ejecutarse directamente, es necesario bajarse un servidor e instalarlo (Tomcat, Jboss o Glassfish), además de tener las licencias oportunas en caso de ser un servidor privado de pago (IBM Websphere).

Visual Studio proporciona una debug propio para poder ir línea a línea en código en el proceso de ejecución.

Sin embargo, en Eclipse es necesario instalarse un “plugin” para poder debuggear el código web. Aunque eclipse también posee un debug propio para clases, es necesario instalarse uno nuevo para el desarrollo web. Uno bueno es el denominado Lomboz válido para debuggear páginas JSP.

En cuanto a requisitos mínimos de cada plataforma, Eclipse es mejor al no necesitar una máquina muy potente para que pueda ejecutarse sin problema. Aunque los requisitos mínimos del Microsoft Visual Studio 2010 son bajos, es necesario tener 2 Gigas de RAM para que no se tengan problemas de lentitud, además cuánto mayor número de núcleos tenga el ordenador será mucho mejor el rendimiento ya que consume mucho. En Eclipse no encontramos este problema y con una máquina que no sea muy potente es posible ejecutarlo sin problema, aunque también depende del servidor que se use. (Por ejemplo Glassfish suele ser más fluido que Jboss) : Nota: Poner los datos de mi PC donde va Eclipse perfecto.

En estos tiempos es muy importante saber cuánto nos vamos a gastar para desarrollar cualquier aplicación. Aquí encontramos una diferencia en la cual una empresa puede elegir una plataforma u otra cambiando el lenguaje de programación. Eclipse tal como se define es un entorno de desarrollo integrado

de código abierto, es decir no hay coste alguno por su uso. Sin embargo para usar legalmente Visual Studio es necesario pagar una licencia de uso que asciende a unos 600 € para desarrolladores.

Una herramienta potente de Eclipse es el control de cambios. Es posible ver los cambios realizados en código visualizándose por fecha y hora. Visual Studio no tiene una herramienta similar instalada, sin embargo es posible añadirle el Microsoft Visual SourceSafe o el Team Foundation Server, el cual su cometido es el control de cambios.



Como se ha comentado anteriormente, en Eclipse es posible instalar números Plugins que pueden llegar hacerle una plataforma muy completa. Incluso se tiene un plugin ("CDT") para poder programar en C o C++. En Visual también se tiene algo parecido de tal forma que se pueden añadir herramientas externas a la plataforma, aunque no es tan potente como en Eclipse.

En cuanto a compatibilidad en sistemas operativos, Eclipse es posible hacer funcionar tanto en sistemas operativos Windows o Unix, mientras que Visual Studio sólo es posible en Windows. En cuanto a arquitectura ambas soportan 32 o 64 bits.

Para acabar nos centramos en la ayuda en línea de dichas plataformas.

11.7.3.2 Cuadro comparativo:

245 Comparativa Eclipse - Microsoft Visual Studio

Plataforma	Coste	Servidor	Requisitos	Debug Web	Control De Cambios	Ayuda En Línea
Eclipse 	Código Abierto	No Incluido	Mínimos	Mediante Plugin	Incluido	Simple
Microsoft Visual Studio 	Pago	Incluido	Mayores	Incluido	No Incluido	Simple

Plataforma	Interfaz	Sistema Operativo	Arquitectura	Multiplataforma
------------	----------	-------------------	--------------	-----------------

Eclipse	Simple	Soporta Windows y Unix	32 y 64 Bits	Si
Microsoft Visual Studio	Simple	Windows	32 y 64 Bits	No

11.7.4 Microsoft Office 2010



Microsoft Office 2010 es la versión más reciente de paquete ofimático de Microsoft.

Cuenta con herramientas para editar textos, realizar hojas de cálculo, presentaciones de diapositivas y otras muchas aplicaciones.

La elaboración de la presente memoria se ha realizado mediante MS Word, para los diagramas UML se ha usado Microsoft Visio 2010, la planificación del proyecto se ha realizado con MS Project y los costes del proyecto se han calculado mediante MS Excel y MS Project.

11.7.5 Microsoft Visio 2010



Herramienta corporativa de Microsoft que permite crear diagramas con plantillas profesionales y modernas formas prediseñadas.

Permite vincular diagramas con orígenes de datos comunes (como Excel) actualizándose automáticamente los datos dentro del diagrama, reflejando elementos visuales llamativos como iconos, símbolos, colores y gráficos de barras.

La simplicidad, las formas basadas en datos y el uso compartido en la Web convierten a Visio 2010 en una herramienta adecuada para gestionar diagramas en nuestro proyecto.

11.7.6 Aris Express



Herramienta gráfica diseñada e implementada para el uso de modelación capaz de guardar y/o exportar los modelos realizados. La principal característica es la utilización de

accesos directos en cada objeto, utilizando conectores, conexiones o similares.

Además permite utilizar estructuras prediseñadas en el área de modelamiento facilitando el trabajo al modelador.

11.7.7 SQL Server 2008



La mejor opción que se puede tener utilizando Windows 7 con IIS 7 es usar otra herramienta integrada de Microsoft para base de datos como es SQL Server 2008.

Ofrece un variado conjunto de características, protección de datos y rendimiento para clientes de aplicaciones incrustadas, aplicaciones web ligeras y almacenes de datos locales.

Proporciona 99,9999% de disponibilidad del tiempo de actividad y mayor seguridad de una de las mejores plataformas de bases de datos.

La integración de Sql Server con código de .Net no resulta costosa, además de que la propia herramienta Visual Studio proporciona un pequeño motor de Sql Server.

Otra opción sería utilizar Mysql, pero Sql Server tiene mayor control de usuarios, cuentas, protección de la base de datos, copias de seguridad, creación de consultas y ejecución de procedimientos almacenados, etc. El gran potencial de Mysql es que la totalidad de sus funcionalidades son gratuitas.

Como última opción usaríamos Oracle, aunque su funcionamiento es similar a Sql Server con pequeñas variaciones en las consultas, por el mismo hecho de seguir usando herramientas corporativas de Microsoft hemos decidido desecharla.

En la siguiente página se muestran los requisitos mínimos:

246 Requisitos Software Sql Server 2008

Requisitos de software mínimos recomendados para SQL Server 2008	
Windows XP (x86)	Service Pack 2, Service Pack 3
Windows Vista (x86 y x64)	Service Pack 1
Windows 7	(x86 y x64)
Windows Server 2003 (x86 y x64)	con Service Pack 2
Windows Server 2003 R2 (x86 y x64)	---
Windows Server 2008 (x86 y x64)	con Service Pack 2
Windows Server 2008 R2 (x64)	---

247 Requisitos Hardware Sql Server 2008

Requisitos de hardware mínimos recomendados para SQL Server 2008		
Arquitectura	32 bits	64 bits
Procesador	1 GHz	1,4 GHz

Memoria RAM	256 Mb de RAM
Disco duro	1 GB de espacio libre

11.7.8 Comparativa Sql Server y Mysql.

248 Comparativa Sql Server Y Mysql

Característica	MySQL	SQL Server Express	SQL Server
Costo	Libre y de pago	Libre	De pago
Open Source	Si	No	No
Plataformas	Linux, Windows y muchas otras	Sólo Windows	Sólo Windows
Limite de tamaño de la base de datos	Limitado por el sistema operativo	10Gb	Limitado por el sistema operativo
Compatibilidad ACID	Depende del motor de almacenamiento	Si	Si
Transacciones	Si	Si	Si
Servicio de reportes	No	Si	Si
Posibilidad de elegir diferentes formas de almacenamiento	Si	No	No
Claves Foráneas	Depende del motor	Si	Si
Vistas	Si	Si	Si
Procedimientos almacenados	Si	Si	Si
Triggers	Si	Si	Si
Cursores	Si	Si	Si
Subconsultas	Si	Si	Si
Replicación	Si	Limitado	Si
Funciones definidas por el usuario (UDF)	Si	Si	Si

12 Conclusiones y trabajos futuros

12.1 Conclusiones

Se han conseguido completamente los objetivos definidos a comienzo de dicho proyecto; se ha desarrollado un estudio del estándar FMS Americano junto con todas sus líneas (MILSTRIP), se ha definido toda una gestión de tiempo y recursos para el completo desarrollo del proyecto y sus fases definiendo para ellos los diagramas de Gantt necesarios junto con las tareas planificadas, se ha llevado a cabo un desarrollo del análisis del sistema y diseño junto con una batería de pruebas necesarias para asegurar que la aplicación funciona correctamente y los requisitos son todos recogidos, y por último también se ha documentado todos los procesos anteriormente definidos creando dicho documento relacionado completamente con la aplicación desarrollada.

He podido comprobar cómo un buen diseño y una buena gestión confirma que cualquier proyecto o aplicación puede ser desarrollado/a sin grandes retrasos.

Además al documentar todo tipo de diseño y análisis, el buscar un componente desarrollado o por ejemplo un requisito de usuario y los requisitos software asociado a él resulta muy sencillo y simple, siempre ahorrando tiempos de búsqueda.

La aplicación de gestión de almacén implementada recoge todos los requisitos definidos. Dicha aplicación es muy manejable de tal forma que cualquier persona no especializada en el sector informático puede usar. Gracias al desarrollo en .Net con el sistema gestor de base de datos Sql Server 2008 he podido aumentar mis conocimientos en dichas tecnologías.

El estudio del estándar FMS es algo que no se ha llevado a cabo públicamente en España y en ningún otro país, por lo que estoy seguro que este proyecto servirá a aquellos que quieran comprender como funciona dicho estándar, además de ser escrito completamente en castellano. Anteriormente a dicho proyecto la única información de dicho estándar sólo es encontrada en los documentos oficiales de EE.UU.

Gracias a la ayuda de mi tutora por parte de la Universidad Belén Ruiz Mecua he podido seguir un índice de todas las tareas que debía realizar,

ayudándome y aconsejándome durante todo el periodo de realización del proyecto.

También he de agradecer a Raúl Díaz por parte de la empresa Sli Uti [<http://www.go2uti.es/>], su dedicación y ayuda en el desarrollo ya que gracias a sus enseñanzas he podido crear una aplicación totalmente definida en una tecnología en la cual no tenía grandes conocimientos.

Por último quiero agradecer dicho proyecto a toda mi familia, cercanos y amigos que me han ayudado en los momentos difíciles y me han felicitado en los mejores momentos.

12.2 Trabajos futuros

Dicho proyecto crea un abanico de posibles inclusiones de funcionalidades y nuevos métodos.

Algunos de ellos son:

- Creación de una aplicación en formato PDA (Java, los, Windows Mobile) con sus correspondientes bases datos y sincronización.
- Inclusión de funcionalidades de localización de mercancías en estanterías, tal como la aplicación Infor WMS 2000 realiza con sus búsquedas por voz.
- Cambio del estándar de funcionamiento, creando versiones que trabajen con distintos formatos, por ejemplo la inclusión del formato EDI (Electronic Data Interchange).
- Sistema de generación de informes de las mercancías consolidadas o modificadas en el almacén, por ejemplo el uso de *Crystal Reports*

13 Resumen en Inglés

This Project consisted to study, develop and test a store manager completely new, whose operation is collected under the American estándar FMS (Foreign Military Sales) .

The Project has been developed in the company Sli Utí [<http://www.go2uti.es/>] under the supervision of Raúl Díaz Gutierrez by the company and Belén Ruiz Mecua by the University Carlos III of Madrid.

First has been developed a Project time management with all it's task, developing a Gantt chart to track and avoid delays.

Tasks has been originated to develop the milestones of this Project with a good control of time and develop.

We have developed a "State of Art" or "State of the Question", evaluating three store manager tools that are in the market today. This has been possible to develop a descriptive tables watching their strengths and weaknesses related to the Software that we will implement the project.

The tasks has been:

- Easy WMS [*Mecalux, Esmena. "Solucione De Almacenaje" [en línea]. December 2011, www.mecalux.es/wms.*]
- GoldenSoft. *"Gestión Comercial Y Contable" [en línea]. December 2011, <http://www.goldensoft.com>*
- Infor Enterprise Software Solutions [*en línea*]. December 2011, www.infor.com

The three tools offers an optimal store manager system with it's integrated database.

The weaknesses regarding this project id that none of them is associated with a standard identification traceability of orders, so it is not possible to use globally.

Besides, all of them are desktop applications, so it's use will be mandatory via installer on the machine itself.

This project does not contemplate such weaknesses as well as a Web application with full access via the Internet, be associated with FMS standard for commercial use with the Government and Ministry of Defense.

Before to start the design tasks we have carried out a catalog of requirements which will consider the application.

The requirements has been divided in:

- User Requirements
 - Capacity
 - Restriction

- System Requirements
 - Functional
 - No functional

- Technical Requirements

This task is an important part ,of this stage depends largely on achieving the ultimate goals as a bad definition can cause that the system doesn't meet all the functionality requested, required and necessary.

Once you have obtained the list of requirements along with their dependency counterfoils ensuring that the requirements are related and there is no ambiguity in its definition or description is passed to the tasks of analysis and system design.

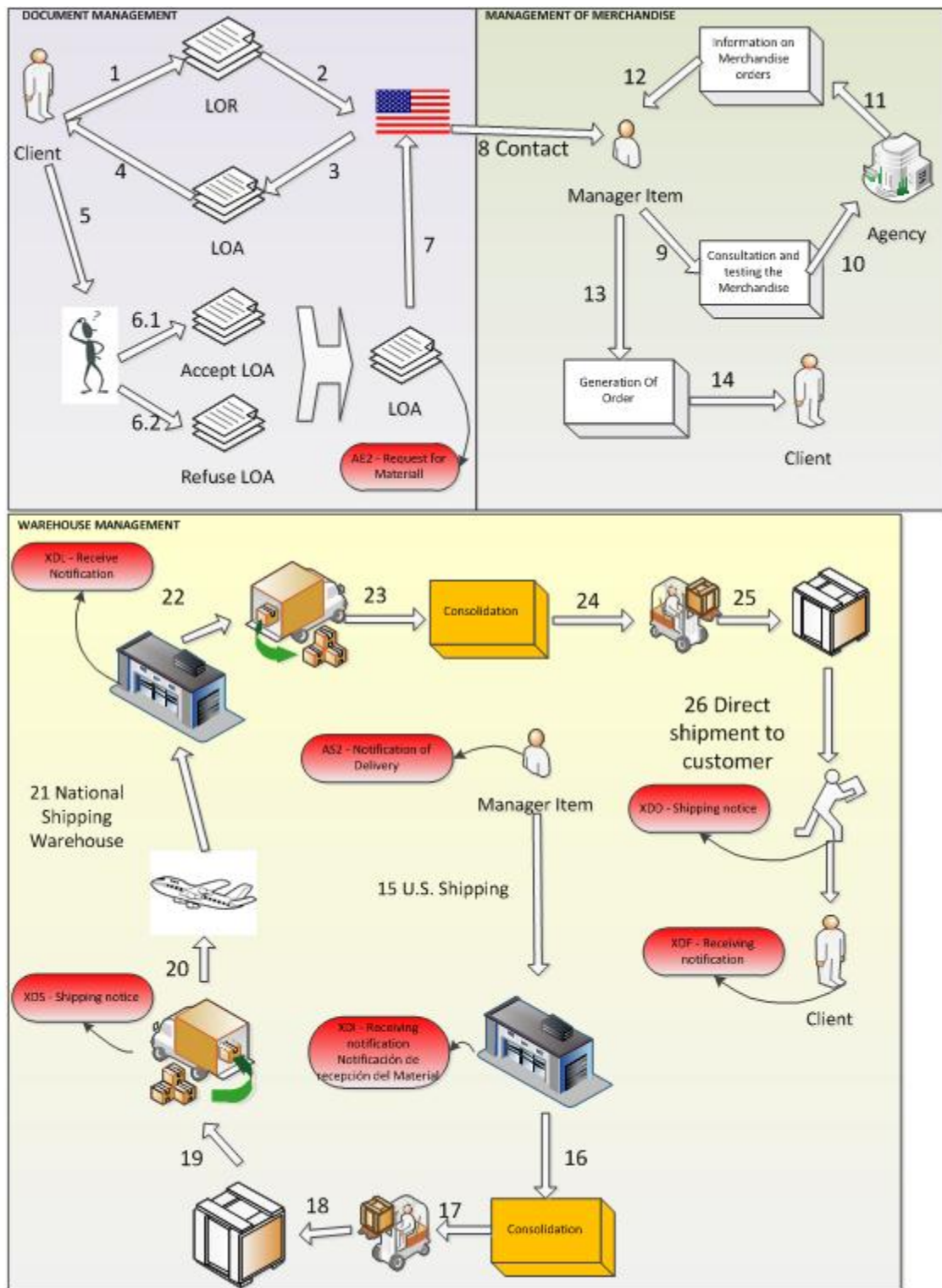
We have developed a comprehensive analysis of the FMS standard operating procedures and all its documentation both at the level of applications.

In this task has been carried out a very precise preparation process model that said host system.

This part is very important to understand what the FMS standard is to develop a software that works in that format.

It has been separations in terms of orders and order not repairable although the large difference in these orders is in shipping by the customer of the goods.

Here we will briefly describe the logic obtained in the buying process an order:



59 Proceso FMS Resumen

The 4 blocks of the analysis are:

1. **Document Management:** Describes the permit application process for submitting a request for a new purchase merchandise or request for repair.

2. **The Merchandise Management:** The process they have to make the armies around the world for the purchase of material, which can be considered as the preparation of the order.

3. **Warehouse Management orders:** The process is performed in the stores from the entrance to the notice of application for a commodity.

4. And finally **the integration of the processes** described above with the global information system FMS.

In the document management:

1. The customer gives the LOR document to USA. This document provides all the materials you want to buy.
2. USA study the LOR recived and check the goods that the customer wants to request. If everything is OK they contact with the control areas responsible for the material and generates the LOA. If USA needs an extra- information by the customer will be communicated to the client before generating the LOA, during, or in the same process or management of the goods.
3. Once it reaches the LOA, the customer can study it and:

-Accept it sending the acceptance with the information neccesary for financing, transport and information.

-Reject. If it's reject the customer can contact again with USA sending a new LOR backing to step 1.

If the customer has accepted the LOA, once it's recived in USA it proceed to consult the stock of each goods and the order generation, so that we enter at the next block.

In the Merchandise Management:

1. Once you have all the information of the goods applied for, contact the "Item Manager", which are responsible for managing the goods with stock number, location and even if they still made or is a good classified in their treatment.
2. The Item Manager is responsible for checking the goods ordered and if any discrepancy occurs generates the order of the goods.
 - If there is no discrepancy and that the good can be served, is the Item

Manager is responsible for the generation of electronic document delivery notice thereof. (Generation of AS2)

In warehouse management, which is where the project manager directly:

1. Once the goods arrives to the store in USA (generation of XDI) is where our store management woks.
2. It takes place the process of consolidation for onward shipment to the store customer who requested the goods. In such consolidation is performed:

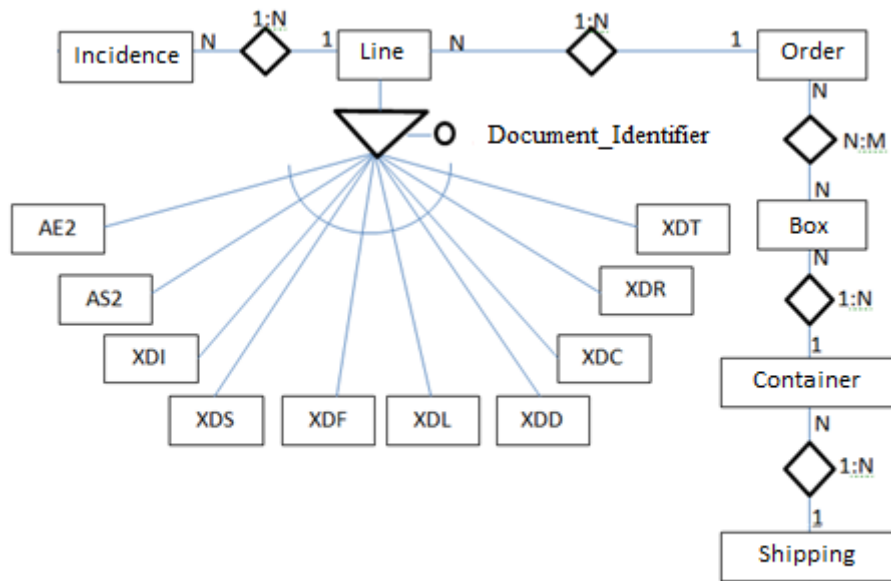
- Shipment Management: The shipments are the biggest containers in which the boxes will consist of those orders. In this management is possible to create a new shipment in the own store, close a shipment to it's send and open a shipment to get the boxes inside.

- Boxes Management: The boxes are smaller than the containers in which goods will order. You can create new boxes from the store, open to enter new orders and close them to be introduced later in a container.

- Consultation orders. Within the manager may consult or take orders for analyzing relevant data can consult them directly orders, orders modified or unassigned pending.

After studying the logic FMS, was carried out performing the functions of the warehouse operator, along with the use cases and the subsequent interaction of objects.

The data model elaboration finished with the data base design and it's entity-relationship diagram.



60 Diagrama E/R Resumen

As reflected in the diagram have the following tables:

- Line: According to the ID it can be of different type.
- Order: The order is associated in one or several boxes.
- Box: is associated with a single cargo.
- Cargo: is associated with a single shipment.
- Shipment: contains one or several cargos.
- Incidence: possible incidences found in a order.

After studied the type of lines that may exist in the logic FMS to the type of properties or attributes contained therein.

In that way the standard is followed faithfully by the store manager system.

Below, we carry out the system design tasks.

We defined the system architecture to see how the store manager should be communicated to the subsequent monitoring of traceability of orders bound him.

The binding and release of FMS lines of the databases is done by Web services- stocking the own database client and systems U.S database to ensure that every order has been sent, received and also can be monitored by other applications that perform this function.

We have conducted classes designed to develop the software along with their methods, attributes and relationships with the physical design of data by creating the database in Sql Server 2008 management system.

In that way we get the class diagram of the application and build scripts for both the database hosted in the U.S. store as the database hosted in the country of origin of the Customer.

Then we carried out a study of possible design alternatives in terms of operating system, database manager, the application format and layout in terms of getting application interfaces why technology selected.

Once the design has defined the next step was to perform the battery of tests to perform on the application and in this way make sure it works properly.

The tests carried out are:

- Unit Tests
 - White box.
 - Black box.
- Integration Tests
- Data insertion tests.

Having already defined tests and requirements linked to their corresponding application has been implemented quickly and that the design has been validated and verified with the requirements.

Interface sample insertion order in boxes:

Universidad Carlos III de Madrid

Principal · Gestión De Caja · Gestión De Cargamento · Gestión De Envíos · Alta Incidencia · Recepciones · Consultas FMS · Consolidación · Salir

Caja: CAJ-00000004

Contenido De La Caja:

Selección Pedido:

Código:

☐ Cerrar Pedido

Pedido	Subjeto	Fecha De Recepción	Ver
EjemploPed	0	14/05/2012 19:08:52	Insertar

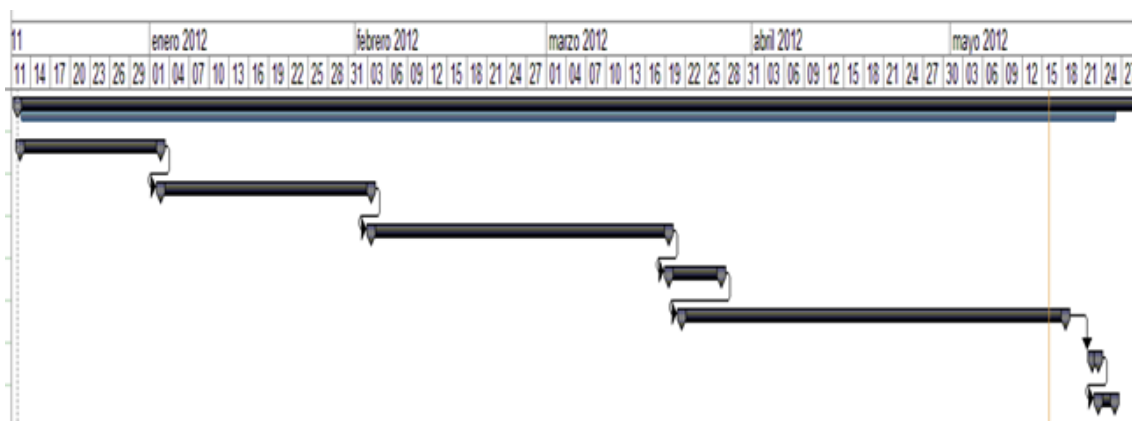
Universidad Carlos III

61 Interfaz Ejemplo Resumen

Finally, we have carried out a development application including implementation of technologies that must be installed and how to do everything to work properly.

We have managed the Gantt chart getting the costs were necessary along with the development time of all tasks, including data we obtain that the total project cost has been € 45,302 all together, VAT included.

As reflected in the following diagram, we see that the project has been conducted for six months, from December 2011 to late May 2012



62 Gantt Resumen

14 Referencias

249 Referencias

Índice	Referencia	Consulta
1	www.dsca.mil	12/12/2011
2	http://es.wikipedia.org/wiki/Windows_7	13/12/2011
3	http://www.iis.net/overview	13/12/2011
4	http://www.microsoft.com/spain/visualstudio/products/2010-editions/professional	13/12/2011
5	http://www.microsoft.com/spain/office/make-it-great/es-es/WhatsNew.aspx	13/12/2011
6	http://emea.microsoftstore.com/es/es-ES/Microsoft/Visio-Standard-2010	13/12/2011
7	http://office.microsoft.com/es-ar/visio/caracteristicas-y-ventajas-de-visio-2010-HA101631752.aspx	13/12/2011
8	http://www.microsoft.com/es-es/sqlserver/product-info/why-sql-server.aspx	13/12/2011
9	http://www.latindevelopers.com/articulos/sql-server/diferencias-entre-mysql-y-sql-server.php	26/12/2011
10	http://es.wikipedia.org/wiki/Windows_7	13/12/2011
11	http://www.iis.net/overview	13/12/2011
12	http://www.microsoft.com/spain/visualstudio/products/2010-editions/professional	13/12/2011
13	http://www.microsoft.com/spain/office/make-it-great/es-es/WhatsNew.aspx	13/12/2011
14	http://emea.microsoftstore.com/es/es-ES/Microsoft/Visio-Standard-2010	13/12/2011
15	http://office.microsoft.com/es-ar/visio/caracteristicas-y-ventajas-de-visio-2010-HA101631752.aspx	13/12/2011
16	http://www.microsoft.com/es-es/sqlserver/product-info/why-sql-server.aspx	13/12/2011
17	www.mecalux.es/wms	15/12/2011
18	http://www.goldensoft.com	16/12/2011
19	www.infor.com	16/12/2011

15 Anexo A Trazabilidad Requisitos Usuario/Software

Usuario/ SW	FSR –01	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
UC-0001	X	X																													
UC-0002					X	X																									
UC-0003							X	X																							
UC-0004									X																						
UC-0005										X	X																				
UC-0006														X																	
UC-0007												X																			
UC-0008															X	X	X														
UC-0009																		X	X												
UC-0010																				X	X										
UC-0011																						X	X								
UC-0012																								X	X	X					
UC-0013																											X	X			
UC-0014																													X	X	X

UC-0015																														
UC-0016																														
UC-0017																														
UC-0018																														
UC-0019	X	X																												
UR-0020																												X	X	
UR-0021																														
UR-0022																														
UR-0023				X																										
UR-0024			X																											
UR-0025																														
UR-0026										X																				
UR-0027																														
UR-0028																											X			

Usuario/ Sw	32	33	34	35	36	37	38	39	40	41	42	43	44
UC-0001													
UC-0002													
UC-0003													
UC-0004													
UC-0005													
UC-0006													
UC-0007													
UC-0008													
UC-0009													
UC-0010													
UC-0011													
UC-0012													
UC-0013													
UC-0014													
UC-0015	X	X	X	X									
UC-0016					X	X							
UC-0017							X	X					
UC-0018									X	X			
UC-0019													
UR-0020			X		X	X	X	X					
UR-0021													
UR-0022												X	
UR-0023													
UR-0024													
UR-0025													X
UR-0026													
UR-0027		X											
UR-0028													

16 Anexo B Glosario

250 Glosario

Término	Descripción
Foreing Military Sales (FMS)	Estándar Americano que precisa el proceso de comprar de una mercancía al ministerio de defensa de EE.UU
The Defense Security Cooperation Agency (DSCA).	Agencia de Cooperación de Seguridad de Defensa
Department of Defense (DoD).	Departamento de Defensa.
Secure Sockets Layer (SSL)	Protocolo de capa de conexión segura
Internet Information Services (IIS)	Servidor de Microsoft
x86	Denominación genérica dada a ciertos microprocesadores de la familia Intel para una arquitectura de 32 bits
x64	Arquitectura basada en la extensión del conjunto de instrucciones x86 para manejar direcciones de 64 bits
Binary digit (Bit)	Un bit es un dígito del sistema de numeración binario
Random-Access Memory (RAM)	Memoria desde donde el procesador recibe las instrucciones y guarda los resultados. La expresión memoria RAM se utiliza frecuentemente para referirse a los módulos de memoria utilizados en los computadores personales y servidores
Aplication Programming Interface (API)	Serie de servicios o funciones que el Sistema Operativo ó librería ofrece al programador
DirectX	Es una colección de APIs desarrolladas para facilitar las complejas tareas relacionadas con multimedia, especialmente programación de juegos y vídeo, en la plataforma Microsoft Windows.
Megabyte (MB)	Unidad de medida de cantidad de datos informáticos. Es un múltiplo del byte u octeto, que equivale a 10^6 bytes

Gigabyte (GB)	Un gigabyte es una unidad de almacenamiento de información, equivale a 10^9 bytes
Gigahercio (GHz)	El gigahercio es un múltiplo de la unidad de medida de frecuencia hercio (Hz) y equivale a 10^9 (1.000.000.000) Hz. Por lo tanto, tiene un ciclo de 1 nanosegundo
RPM	Revoluciones por minuto es una unidad de frecuencia, usada frecuentemente para medir la velocidad angular
DVD-ROM	Es un disco con la capacidad de ser utilizado para leer o reproducir datos o información (audio, imágenes, video, texto, etc), es decir, puede contener diferentes tipos de contenido como películas cinematográficas, videojuegos, datos, música, etc.Almacena 4,7 Gb según los fabricantes en base decimal, y aproximadamente 4,377 Gb reales en base binaria o Gb de datos en una cara del disco.
Service Pack	consiste en un grupo de parches que actualizan, corrigen y mejoran aplicaciones y sistemas operativos
Mysql	Sistema de gestión de bases de datos relacional desarrollado por Sun Microsystems (2008)y ésta a su vez por Oracle Corporation (2009) como Software libre.
Oracle	sistema de gestión de base de datos relacional desarrollado por Oracle Corporation
Java2E ó JavaEE	proporciona una estructura de ejecución llamada MVC (Java Virtual Machine), conocida como entorno de desarrollo Java, que es la base del J2EE y de las tecnologías Java Web Services
Eclipse Enterprise	Entorno de desarrollo integrado de código abierto multiplataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados como el IDE de Java llamado Java Development Toolkit.
Jboss	Servidor de aplicaciones J2EE de código abierto implementado en Java puro.
Tomcat	También llamado Jakarta Tomcat funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation
Glassfish	Servidor de aplicaciones de software libre desarrollado por Sun Microsystems
IBM Websphere	Middleware diseñado para configurar, operar e integrar aplicaciones de e-business a través de varias plataformas de red usando las tecnologías del Web.

Plugin (Complemento)	Aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.
Tracking	Rastreo de seguimiento para acceder a la mercancía solicitada
Palet	Bandeja de carga que soporta los embalajes y los constituye en una unidad de carga.
LOR	Documentación oficial de solicitud de pedidos.
LOA	Documentación oficial del estado de la solicitud de pedidos.
DAMES	Aplicación, Software propio de EE.UU para la actualización de las líneas propias de los clientes bajo el formato FMS.
DAASC	Agencia de logística oficial de EE.UU
TRIL	Repositorio en el cual se detallan los materiales reparables bajo el formato FMS.
AE2	Línea bajo el formato FMS que detalla el proceso de petición del material.
AS2	Línea bajo el formato FMS que notifica el envío de material al almacén de EE.UU
XDR	Línea bajo el formato FMS que notifica la recepción del material en el almacén de EE.UU
XDS	Línea bajo el formato FMS que notifica la consolidación y envío del material desde el almacén de EE.UU
XDL	Línea bajo el formato FMS que notifica la recepción del material en el almacén del cliente.
XDD	Línea bajo el formato FMS que notifica el envío del material en el almacén del cliente.
XDF	Línea bajo el formato FMS que notifica la recepción del material por parte del cliente.
XDC	Línea bajo el formato FMS que notifica la recepción del material reparable para su posterior envío en el almacén del cliente.
XDR	Línea bajo el formato FMS que notifica la recepción del material reparable para su posterior envío en el almacén del cliente
XDT	Línea bajo el formato FMS que notifica el envío del material reparable al almacén de EE.UU

17 Anexo C Script Creación Base De Datos EE.UU

La base de datos creada deberá llamarse AlmacenFMS.mdf

```
USE [AlmacenFMS]
GO
/***** Object: Table [dbo].[Usuarios]  Script Date: 05/10/2012 19:11:00 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Usuarios](
    [Dni] [nvarchar](9) NOT NULL,
    [Nombre] [nvarchar](20) NOT NULL,
    [Contrasenia] [nvarchar](50) NOT NULL,
    [Fecha_Creacion] [datetime] NOT NULL,
    [Estado] [bit] NOT NULL,
    [Tipo] [nvarchar](15) NOT NULL,
    [Fecha_Caducidad] [datetime] NULL,
    CONSTRAINT [PK_Usuarios_1] PRIMARY KEY CLUSTERED
(
    [Dni] ASC,
    [Nombre] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Envio]  Script Date: 05/10/2012 19:11:01 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Envio](
    [Identificador] [nvarchar](20) NOT NULL,
    [Enviado] [bit] NULL,
    [Tipo] [nvarchar](20) NULL,
    [Tracking] [nvarchar](50) NULL,
    [Fecha_Creacion] [datetime] NULL,
    CONSTRAINT [PK_Envio] PRIMARY KEY CLUSTERED
(
    [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Pedido]  Script Date: 05/10/2012 19:11:01 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Pedido](
    [Numero_Documento] [nvarchar](13) NOT NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
    [Consolidado] [bit] NULL,
    [Fecha_Consolidacion] [datetime] NULL,
    [Fecha_Recepcion] [datetime] NULL,
```

```

        [Fecha_Estimada] [datetime] NULL,
CONSTRAINT [PK_Pedido_1] PRIMARY KEY CLUSTERED
(
        [Numero_Documento] ASC,
        [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Linea]  Script Date: 05/10/2012 19:11:01 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Linea](
        [Identificador] [nvarchar](4) NOT NULL,
        [Numero_Documento] [nvarchar](13) NOT NULL,
        [Linea] [nvarchar](300) NOT NULL,
        [Fecha_Entrada] [datetime] NULL,
        [Sufijo] [nvarchar](2) NOT NULL,
CONSTRAINT [PK_Linea_1] PRIMARY KEY CLUSTERED
(
        [Identificador] ASC,
        [Numero_Documento] ASC,
        [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Incidencias]  Script Date: 05/10/2012 19:11:01 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Incidencias](
        [Numero] [int] IDENTITY(1,1) NOT NULL,
        [Pedido] [nvarchar](13) NOT NULL,
        [Sufijo] [nvarchar](2) NOT NULL,
        [Tipo] [nvarchar](20) NOT NULL,
        [Observaciones] [nvarchar](150) NOT NULL,
        [Usuario] [nvarchar](20) NOT NULL,
        [Fecha_Creacion] [datetime] NULL,
CONSTRAINT [PK_Incidencias] PRIMARY KEY CLUSTERED
(
        [Numero] ASC,
        [Pedido] ASC,
        [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Habilitar_Usuario]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Habilitar_Usuario]
        @Usuario nvarchar(20)

AS
        update Usuarios
        set Estado = 1
        where Nombre = @Usuario

```



```

        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_EliminarUsuario]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_EliminarUsuario]

        @Usuario nvarchar(20)

AS

        delete from Usuarios where Nombre = @Usuario
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Deshabilitar_Usuario]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
create PROCEDURE [dbo].[sp_Deshabilitar_Usuario]

        @Usuario nvarchar(20)

AS

        update Usuarios
        set Estado = 0
        where Nombre = @Usuario
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Cambiar_Contrasenia]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cambiar_Contrasenia]

        @Usuario nvarchar(20),
        @Contrasenia nvarchar(50),
        @dni nvarchar(9),
        @FechaCaducidad datetime

AS

        update Usuarios set
        Contrasenia = @Contrasenia,
        Fecha_Caducidad = @FechaCaducidad
        where Nombre =@Usuario and
        Dni = @dni
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Usuario]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Usuario]

        @Usuario nvarchar(20),
        @Contrasenia nvarchar(50),
        @Dni nvarchar(9),
        @Tipo nvarchar(15),
        @Fecha_Creacion datetime,
        @Fecha_Caducidad datetime

AS

```

```

        Insert into Usuarios(Dni,Nombre, Contrasenía,Fecha_Creacion,Estado,Tipo,Fecha_Caducidad)

        values(@Dni,@Usuario,@Contrasenía,@Fecha_Creacion,1,@Tipo,@Fecha_Caducidad)
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Pedido]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Pedido]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2),
    @IdentificadorLinea nvarchar(3),
    @Fecha_Recepcion datetime,
    @Fecha_Estimada datetime
AS
    Insert into Pedido(Numero_Documento,Sufijo,Fecha_Recepcion,Fecha_Estimada)
        values (@Pedido,@Sufijo,@Fecha_Recepcion,@Fecha_Estimada)

    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Linea]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Linea]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2),
    @Linea nvarchar(300),
    @IdentificadorLinea nvarchar(4),
    @Fecha_Entrada datetime
AS
    insert into Linea (Identificador, Numero_Documento,Linea,Fecha_Entrada,Sufijo)
        values(@IdentificadorLinea,@Pedido,@Linea,@Fecha_Entrada,@Sufijo)
    RETURN

GO
/***** Object:  StoredProcedure  [dbo].[sp_Actualizar_Envio_Tracking]  Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Actualizar_Envio_Tracking]

    @IdentificadorEnvio nvarchar(20),
    @Tracking nvarchar(50)
AS
    update Envio set Tracking = @Tracking, Enviado = 1 where Identificador = @IdentificadorEnvio
    RETURN

GO
/***** Object: Table [dbo].[Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Cargamento](
    [Identificador] [nvarchar](20) NOT NULL,
    [Envio] [nvarchar](20) NULL,
    [Fecha_Creacion] [datetime] NOT NULL,
    [Estado] [bit] NULL,

```

```

        [Peso] [float] NULL,
        [Volumen] [float] NULL,
        [Altura] [float] NULL,
        [Anchura] [nchar](10) NULL,
        [Tipo] [nvarchar](20) NULL,
CONSTRAINT [PK_Cargamento] PRIMARY KEY CLUSTERED
(
        [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Enviados]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Enviados]

        @Pedido nvarchar(13)

AS
        select * from Linea where Numero_Documento like @Pedido + '%'and Identificador ='XDS'
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Linea]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Linea]

        @Pedido nvarchar(20),
        @Sufijo nvarchar(2),
        @IdentificadorLinea nvarchar(4)

AS
        select * from Linea where Identificador = @IdentificadorLinea and Numero_Documento = @Pedido and
Sufijo = @Sufijo
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Envio]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Envio]

        @IdentificadorEnvio nvarchar(20),
        @TipoEnvio nvarchar(20),
        @Fecha_Creacion datetime

AS
        insert into Envio (Identificador, Tipo, Fecha_Creacion, Enviado)
values (@IdentificadorEnvio, @TipoEnvio, @Fecha_Creacion, 0)
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Login_Usuario]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Login_Usuario]

```

```

        @Usuario nvarchar(20)

AS
    select * from Usuarios where Nombre = @Usuario
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Identificador_Linea]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Identificador_Linea]
    @Pedido nvarchar(13),
    @Sufijo nvarchar(2)

AS
    select Identificador, fecha_Entrada from Linea where Numero_Documento = @Pedido and Sufijo = @Sufijo
    order by fecha_Entrada desc

    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_EnviosNoConsolidados]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_EnviosNoConsolidados]

    @IdentificadorEnvio nvarchar(20)

AS
    select * from Envio where enviado =0 and Identificador like @IdentificadorEnvio + '%'
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Envios_Busq]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Envios_Busq]

    @IdentificadorEnvio nvarchar(20)

AS
    select *from Envio where Identificador like @IdentificadorEnvio + '%'
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Envio]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Envio]

    @IdentificadorEnvio nvarchar(20)

AS
    select * from Envio where Identificador = @IdentificadorEnvio
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Todos]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Todos]

@Pedido nvarchar(13)
AS
    select * from Pedido where Numero_Documento like @Pedido + '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Pendientes]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Pendientes]
    @Pedido nvarchar(13)
AS
SELECT      Linea.Identificador, Linea.Numero_Documento, Linea.Linea, Linea.Fecha_Entrada, Linea.Sufijo
FROM        Linea LEFT OUTER JOIN
            Pedido ON Linea.Numero_Documento = Pedido.Numero_Documento AND Linea.Sufijo = Pedido.Sufijo
WHERE       (Pedido.Numero_Documento IS NULL) and Linea.Numero_Documento like @Pedido + '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_NoProcesados]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_NoProcesados]

    @Pedido nvarchar(13)
AS
    select * from Pedido where Numero_Documento like @Pedido + '%' and Consolidado =0
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Usuarios]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Usuarios]
    @Usuario nvarchar(20)
AS
    select * from Usuarios where Nombre like @Usuario+ '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Usuario_ByNombre]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Usuario_ByNombre]

    @Usuario nvarchar(20)

```

```

AS
    select * from Usuarios where Nombre = @Usuario
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimoEnvio]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_UltimoEnvio]
    /*
    (
        @parameter1 int = 5,
        @parameter2 datatype OUTPUT
    )
    */
AS
    SELECT      TOP (1) Identificador, Tipo, Fecha_Creacion AS Expr1
    FROM        Envio
    ORDER BY Expr1 DESC

RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimoCargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_UltimoCargamento]
    /*
    (
        @parameter1 int = 5,
        @parameter2 datatype OUTPUT
    )
    */
AS
    SELECT      TOP (1) Identificador, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura, Tipo,
    Fecha_Creacion AS Expr1
    FROM        Cargamento
    ORDER BY Expr1 DESC

RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Detalle_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Detalle_Cargamento]
    @IdentificadorCargamento nvarchar(20)
AS
    select * from Cargamento where Identificador = @IdentificadorCargamento
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Contenido_Envio]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Contenido_Envio]

```

```

        @IdentificadorEnvio nvarchar(20)

AS
SELECT      Cargamento.*
FROM        Cargamento INNER JOIN
            Envio ON Cargamento.Envio = Envio.Identificador and Cargamento.Envio = @IdentificadorEnvio
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Cargamento]

    @identificador nvarchar(20),
    @tipo nvarchar(20),
    @peso float,
    @volumen float,
    @altura float,
    @anchura float,
    @fecha_Creacion datetime,
    @Estado bit

AS

Insert into Cargamento(identificador,Tipo,Fecha_Creacion,Estado,Peso,Volumen,Altura,Anchura)

values(@identificador,@tipo,@fecha_Creacion,@Estado,@peso,@volumen,@altura,@anchura)
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Cargamento_En_Envio]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Cargamento_En_Envio]
    @IdentificadorEnvio nvarchar(20),
    @IdentificadorCargamento nvarchar(20)

AS
update Cargamento set Envio = @IdentificadorEnvio where Identificador = @IdentificadorCargamento
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cargamentos_NoInsertados]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cargamentos_NoInsertados]

    @IdentificadorCargamento nvarchar(20)

AS

SELECT      Identificador, Envio, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura, Tipo
FROM        Cargamento
WHERE       (Envio IS NULL) and Identificador like @IdentificadorCargamento + '%' and Cargamento.Estado = 0
RETURN

GO

```

```

/***** Object: StoredProcedure [dbo].[sp_Obtener_Cargamentos]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cargamentos]

    @Identificador nvarchar(20),
    @Estado nvarchar(20)

AS
if @Estado = 'Abiertos'
begin
    select * from Cargamento where Identificador like @Identificador + '%' and Estado =1
end

if @Estado = 'Cerrados'
begin
    select * from Cargamento where Identificador like @Identificador + '%' and Estado = 0
end

if @Estado = 'Todos'
begin
    select * from Cargamento
end
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Incidencias]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Incidencias]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2)

AS
    select * from Incidencias where Pedido = @Pedido and Sufijo = @Sufijo
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Cargamento_Pedido]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Sacar_Cargamento_Pedido]

    @IdentificadorEnvio nvarchar(20),
    @IdentificadorCargamento nvarchar(20)

AS
    Update Cargamento set Envio = null where Identificador =@IdentificadorCargamento
RETURN
GO
/***** Object: Table [dbo].[Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Caja](
    [Identificador] [nvarchar](20) NOT NULL,
    [Cargamento] [nvarchar](20) NULL,
    [Fecha_Creacion] [datetime] NOT NULL,

```



```

        [Estado] [bit] NULL,
        [Peso] [float] NULL,
        [Volumen] [float] NULL,
        [Altura] [float] NULL,
        [Anchura] [float] NULL,
        [Tipo] [nvarchar](20) NULL,
        [Descripcion] [nvarchar](150) NULL,
CONSTRAINT [PK_Caja] PRIMARY KEY CLUSTERED
(
        [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Abrir_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Abrir_Cargamento]

        @Identificador nvarchar(20)
AS
        update Cargamento set Estado = 1 where Identificador = @Identificador
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Incidencia]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Incidencia]
@Pedido nvarchar(13),
@Sufijo nvarchar(2),
@tipoIncidencia nvarchar(20),
@observaciones nvarchar(150),
@usuario nvarchar(20),
@fecha_Creacion datetime
AS
        insert into Incidencias (Pedido,Sufijo,Tipo,Observaciones,Usuario,Fecha_Creacion)
        values (@Pedido,@Sufijo,@tipoIncidencia,@observaciones,@usuario,@fecha_Creacion)
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Cerrar_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cerrar_Cargamento]

        @Identificador nvarchar(20)
AS
        update Cargamento set Estado = 0 where Identificador = @Identificador
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Cerrar_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cerrar_Caja]

        @Identificador nvarchar(20)

```

```

AS
    update Caja set Estado = 0 where Identificador = @Identificador
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Abrir_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Abrir_Caja]

    @Identificador nvarchar(20)

AS
    update Caja set Estado = 1 where Identificador = @Identificador
RETURN
GO
/***** Object: Table [dbo].[Pedido_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Pedido_Caja](
    [Caja] [nvarchar](20) NOT NULL,
    [Pedido] [nvarchar](13) NOT NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
    [Fecha_Inserccion] [datetime] NULL,
    [DivisionesTotales] [int] NULL,
    [NumeroDivision] [int] NOT NULL,
CONSTRAINT [PK_Pedido_Caja] PRIMARY KEY CLUSTERED
(
    [Caja] ASC,
    [Pedido] ASC,
    [Sufijo] ASC,
    [NumeroDivision] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimaCaja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_UltimaCaja]
/*
(
    @parameter1 int = 5,
    @parameter2 datatype OUTPUT
)
*/
AS
    SELECT      TOP (1) Identificador, Cargamento, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura,
Tipo, Descripcion, Fecha_Creacion AS Expr1
    FROM        Caja
    ORDER BY Expr1 DESC

RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```

CREATE PROCEDURE [dbo].[sp_Sacar_Caja]

    @IdentificadorCargamento nvarchar(20),
    @IdentificadorCaja nvarchar(20)
AS
    update Caja set Cargamento =null where Identificador = @IdentificadorCaja
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cajas_NoInsertadas]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cajas_NoInsertadas]
    @IdentificadorCaja nvarchar(20)

AS
    SELECT          Caja.Identificador, Caja.Cargamento, Caja.Fecha_Creacion, Caja.Estado, Caja.Peso,
Caja.Volumen, Caja.Altura, Caja.Anchura, Caja.Tipo, Caja.Descripcion
FROM          Caja LEFT OUTER JOIN
                Cargamento ON Caja.Cargamento = Cargamento.Identificador
WHERE          (Caja.Estado = 0) AND (Cargamento.Identificador IS NULL) and Caja.Identificador like
@IdentificadorCaja + '%'
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cajas]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cajas]

    @Identificador nvarchar(20),
    @Estado nvarchar(20)

AS
if @Estado = 'Abiertas'
    begin
        select * from Caja where Identificador like @Identificador + '%' and Estado =1
    end

if @Estado = 'Cerradas'
    begin
        select * from Caja where Identificador like @Identificador + '%' and Estado = 0
    end

if @Estado = 'Todas'
    begin
        select * from Caja
    end
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Caja]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Caja]

    @IdentificadorCargamento nvarchar(20),
    @IdentificadorCaja nvarchar(20)

```

```

AS
    update Caja set Cargamento = @IdentificadorCargamento where Identificador = @IdentificadorCaja
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Caja]

    @identificador nvarchar(20),
    @tipo nvarchar(20),
    @peso float,
    @volumen float,
    @altura float,
    @anchura float,
    @descripcion nvarchar(150),
    @fecha_Creacion datetime,
    @Estado bit

AS

    Insert into Caja(identificador,Tipo,Fecha_Creacion,Estado,Peso,Volumen,Altura,Anchura,Descripcion)

    values(@identificador,@tipo,@fecha_Creacion,@Estado,@peso,@volumen,@altura,@anchura,@descripcion)

RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Contenido_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Contenido_Cargamento]
@Identificador nvarchar(20)

AS

    select * from Caja where Cargamento = @Identificador
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Detalle_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Detalle_Caja]

    @IdentificadorCaja nvarchar(20)

AS

    select * from Caja where Identificador = @IdentificadorCaja
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Pedido_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Sacar_Pedido_Caja]

```

```

        @IdentificadorPedido nvarchar(13),
        @Sufijo nvarchar(2)
AS
    delete from Pedido_Caja where Pedido = @IdentificadorPedido and Sufijo = @Sufijo
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Ultimo_PedidoDiv]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Ultimo_PedidoDiv]

    @IdentificadorPedido nvarchar(13)

AS
    select * from Pedido_Caja where Pedido = @IdentificadorPedido order by NumeroDivision DESC
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_NoInsertados]    Script Date: 05/10/2012 19:11:05
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_NoInsertados]

    @Pedido nvarchar(13)

AS
SELECT          Pedido.Numero_Documento, Pedido.Sufijo, Pedido.Consolidado, Pedido.Fecha_Consolidacion,
Pedido.Fecha_Recepcion, Pedido.Fecha_Estimada

FROM            Pedido LEFT OUTER JOIN
                Pedido_Caja ON    Pedido.Sufijo =    Pedido_Caja.Sufijo    AND    Pedido_Caja.Pedido =
Pedido.Numero_Documento

WHERE           (Pedido_Caja.Pedido IS NULL) and Pedido.Numero_Documento like @Pedido + '%'
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Envio]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Envio]

    @IdentificadorEnvio nvarchar(20)

AS
    SELECT      Pedido.Numero_Documento, Pedido.Sufijo, Pedido.Consolidado, Pedido.Fecha_Consolidacion,
Pedido.Fecha_Recepcion, Pedido.Fecha_Estimada
FROM          Caja INNER JOIN
              Cargamento ON Caja.Cargamento = Cargamento.Identificador INNER JOIN
              Envio ON Cargamento.Envio = Envio.Identificador INNER JOIN
              Pedido_Caja ON Caja.Identificador = Pedido_Caja.Caja INNER JOIN
              Pedido ON Pedido_Caja.Pedido = Pedido.Numero_Documento AND Pedido_Caja.Sufijo = Pedido.Sufijo
WHERE         (Cargamento.Envio = @IdentificadorEnvio)
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Datos_Caja]    Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON

```

```

GO
CREATE PROCEDURE [dbo].[sp_Obtener_Datos_Caja]

    @IdentificadorCaja nvarchar(20)

AS
    SELECT          Caja.Identificador, Caja.Cargamento, Caja.Fecha_Creacion, Caja.Estado, Caja.Peso,
    Caja.Volumen, Caja.Altura, Caja.Anchura, Caja.Tipo, Caja.Descripcion,
                    Pedido_Caja.Caja,    Pedido_Caja.Pedido,    Pedido_Caja.Sufijo,    Pedido_Caja.Fecha_Inserccion,
    Pedido_Caja.DivisionesTotales, Pedido_Caja.NumeroDivision
FROM      Caja INNER JOIN
          Pedido_Caja ON Caja.Identificador = Pedido_Caja.Caja
WHERE     (Pedido_Caja.Caja = @IdentificadorCaja)
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Pedido_En_Caja]  Script Date: 05/10/2012 19:11:05 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Pedido_En_Caja]

    @IdentificadorCaja nvarchar(20),
    @IdentificadorPedido nvarchar(50),
    @IdentificadorSufijo nvarchar(2),
    @Fecha_Inserccion datetime,
    @NumeroTotalPartes int,
    @Parte int

AS
    Insert Into Pedido_Caja (Caja, Pedido,Sufijo,Fecha_Inserccion,DivisionesTotales,NumeroDivision)

    values(@IdentificadorCaja,@IdentificadorPedido,@IdentificadorSufijo,@Fecha_Inserccion
    ,@NumeroTotalPartes,@Parte)
    RETURN

GO
/***** Object: ForeignKey [FK_Incidencias_Pedido]  Script Date: 05/10/2012 19:11:01 *****/
ALTER TABLE [dbo].[Incidencias] WITH CHECK ADD CONSTRAINT [FK_Incidencias_Pedido] FOREIGN KEY([Pedido],
[Sufijo])
REFERENCES [dbo].[Pedido] ([Numero_Documento], [Sufijo])
GO
ALTER TABLE [dbo].[Incidencias] CHECK CONSTRAINT [FK_Incidencias_Pedido]
GO
/***** Object: ForeignKey [FK_Cargamento_Envio]  Script Date: 05/10/2012 19:11:05 *****/
ALTER TABLE [dbo].[Cargamento] WITH CHECK ADD CONSTRAINT [FK_Cargamento_Envio] FOREIGN KEY([Envio])
REFERENCES [dbo].[Envio] ([Identificador])
GO
ALTER TABLE [dbo].[Cargamento] CHECK CONSTRAINT [FK_Cargamento_Envio]
GO
/***** Object: ForeignKey [FK_Caja_Cargamento]  Script Date: 05/10/2012 19:11:05 *****/
ALTER TABLE [dbo].[Caja] WITH CHECK ADD CONSTRAINT [FK_Caja_Cargamento] FOREIGN KEY([Cargamento])
REFERENCES [dbo].[Cargamento] ([Identificador])
GO
ALTER TABLE [dbo].[Caja] CHECK CONSTRAINT [FK_Caja_Cargamento]
GO
/***** Object: ForeignKey [FK_Pedido_Caja_Caja]  Script Date: 05/10/2012 19:11:05 *****/
ALTER TABLE [dbo].[Pedido_Caja] WITH CHECK ADD CONSTRAINT [FK_Pedido_Caja_Caja] FOREIGN KEY([Caja])
REFERENCES [dbo].[Caja] ([Identificador])
GO
ALTER TABLE [dbo].[Pedido_Caja] CHECK CONSTRAINT [FK_Pedido_Caja_Caja]
GO
/***** Object: ForeignKey [FK_Pedido_Caja_Pedido1]  Script Date: 05/10/2012 19:11:05 *****/
ALTER TABLE [dbo].[Pedido_Caja] WITH CHECK ADD CONSTRAINT [FK_Pedido_Caja_Pedido1] FOREIGN
KEY([Pedido], [Sufijo])

```

```
REFERENCES [dbo].[Pedido] ([Numero_Documento], [Sufijo])
GO
ALTER TABLE [dbo].[Pedido_Caja] CHECK CONSTRAINT [FK_Pedido_Caja_Pedido1]
```

18 Anexo D Script Creación Base De Datos Cliente

La base de datos creada deberá llamarse AlmacenFMSCiente.mdf

```
USE [AlmacenFMSCiente]
GO
/***** Object: Table [dbo].[Usuarios]  Script Date: 05/14/2012 18:52:44 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Usuarios](
    [Dni] [nvarchar](9) NOT NULL,
    [Nombre] [nvarchar](20) NOT NULL,
    [Contrasenia] [nvarchar](50) NOT NULL,
    [Fecha_Creacion] [datetime] NOT NULL,
    [Estado] [bit] NOT NULL,
    [Tipo] [nvarchar](15) NOT NULL,
    [Fecha_Caducidad] [datetime] NULL,
    CONSTRAINT [PK_Usuarios] PRIMARY KEY CLUSTERED
(
    [Dni] ASC,
    [Nombre] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Envio]  Script Date: 05/14/2012 18:52:44 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Envio](
    [Identificador] [nvarchar](20) NOT NULL,
    [Enviado] [bit] NULL,
    [Tipo] [nvarchar](20) NULL,
    [Tracking] [nvarchar](50) NULL,
    [Fecha_Creacion] [datetime] NULL,
    CONSTRAINT [PK_Envio] PRIMARY KEY CLUSTERED
(
    [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Pedido]  Script Date: 05/14/2012 18:52:44 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Pedido](
    [Numero_Documento] [nvarchar](13) NOT NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
    [Consolidado] [bit] NULL,
```

```

        [Fecha_Consolidacion] [datetime] NULL,
        [Fecha_Recepcion] [datetime] NULL,
        [Fecha_Estimada] [datetime] NULL,
CONSTRAINT [PK_Pedido_1] PRIMARY KEY CLUSTERED
(
    [Numero_Documento] ASC,
    [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Linea]  Script Date: 05/14/2012 18:52:44 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Linea](
    [Identificador] [nvarchar](4) NOT NULL,
    [Numero_Documento] [nvarchar](13) NOT NULL,
    [Linea] [nvarchar](300) NOT NULL,
    [Fecha_Entrada] [datetime] NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
CONSTRAINT [PK_Linea_1] PRIMARY KEY CLUSTERED
(
    [Identificador] ASC,
    [Numero_Documento] ASC,
    [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Incidencias]  Script Date: 05/14/2012 18:52:44 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Incidencias](
    [Numero] [int] IDENTITY(1,1) NOT NULL,
    [Pedido] [nvarchar](13) NOT NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
    [Tipo] [nvarchar](20) NOT NULL,
    [Observaciones] [nvarchar](150) NOT NULL,
    [Usuario] [nvarchar](50) NOT NULL,
    [Fecha_Creacion] [datetime] NULL,
CONSTRAINT [PK_Incidencias] PRIMARY KEY CLUSTERED
(
    [Numero] ASC,
    [Pedido] ASC,
    [Sufijo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Habilitar_Usuario]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Habilitar_Usuario]
    @Usuario nvarchar(20)
AS
    update Usuarios

```



```

        set Estado = 1
        where Nombre = @Usuario
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_EliminarUsuario]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_EliminarUsuario]

    @Usuario nvarchar(20)

AS
    delete from Usuarios where Nombre = @Usuario
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Deshabilitar_Usuario]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
create PROCEDURE [dbo].[sp_Deshabilitar_Usuario]

    @Usuario nvarchar(20)

AS
    update Usuarios
    set Estado = 0
    where Nombre = @Usuario
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Cambiar_Contrasenia]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cambiar_Contrasenia]

    @Usuario nvarchar(20),
    @Contrasenia nvarchar(50),
    @dni nvarchar(9),
    @FechaCaducidad datetime

AS
    update Usuarios set
    Contrasenias = @Contrasenia,
    Fecha_Caducidad = @FechaCaducidad
    where Nombre = @Usuario and
    Dni = @dni
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Usuario]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Usuario]

    @Usuario nvarchar(20),
    @Contrasenia nvarchar(50),
    @Dni nvarchar(9),
    @Tipo nvarchar(15),
    @Fecha_Creacion datetime,
    @Fecha_Caducidad datetime

```

```

AS
    Insert into Usuarios(Dni,Nombre, Contrasenía,Fecha_Creacion,Estado,Tipo,Fecha_Caducidad)

    values(@Dni,@Usuario,@Contrasenía,@Fecha_Creacion,1,@Tipo,@Fecha_Caducidad)
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Pedido]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Pedido]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2),
    @IdentificadorLinea nvarchar(3),
    @Fecha_Recepcion datetime,
    @Fecha_Estimada datetime
AS
    Insert into Pedido(Número_Documento,Sufijo,Fecha_Recepcion,Fecha_Estimada)
        values (@Pedido,@Sufijo,@Fecha_Recepcion,@Fecha_Estimada)

    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Linea]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Linea]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2),
    @Linea nvarchar(300),
    @IdentificadorLinea nvarchar(4),
    @Fecha_Entrada datetime
AS
    insert into Linea (Identificador, Número_Documento,Linea,Fecha_Entrada,Sufijo)
    values(@IdentificadorLinea,@Pedido,@Linea,@Fecha_Entrada,@Sufijo)
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Actualizar_Envio_Tracking]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Actualizar_Envio_Tracking]

    @IdentificadorEnvio nvarchar(20),
    @Tracking nvarchar(50)
AS
    update Envio set Tracking = @Tracking, Enviado = 1 where Identificador = @IdentificadorEnvio
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Actualizar_Envio_Consolidado]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Actualizar_Envio_Consolidado]

    @Pedido nvarchar(13),

```

```

        @Sufijo nvarchar(2),
        @Fecha_Consolidado datetime
AS
    update Pedido set Fecha_Consolidacion =@Fecha_Consolidado, Consolidado = 1 where
Numero_Documento =@Pedido and Sufijo = @Sufijo
    RETURN
GO
/***** Object: Table [dbo].[Cargamento] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Cargamento](
    [Identificador] [nvarchar](20) NOT NULL,
    [Envio] [nvarchar](20) NULL,
    [Fecha_Creacion] [datetime] NOT NULL,
    [Estado] [bit] NULL,
    [Peso] [float] NULL,
    [Volumen] [float] NULL,
    [Altura] [float] NULL,
    [Anchura] [nchar](10) NULL,
    [Tipo] [nvarchar](20) NULL,
    CONSTRAINT [PK_Cargamento] PRIMARY KEY CLUSTERED
(
    [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Enviados] Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Enviados]

    @Pedido nvarchar(13)

AS
    select * from Linea where Numero_Documento like @Pedido + '%and Identificador ='XDD' or
Identificador ='XDT'
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Linea] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Linea]

    @Pedido nvarchar(20),
    @Sufijo nvarchar(2),
    @IdentificadorLinea nvarchar(4)

AS
    select * from Linea where Identificador = @IdentificadorLinea and Numero_Documento = @Pedido and
Sufijo = @Sufijo
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Envio] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON

```

```

GO
CREATE PROCEDURE [dbo].[sp_Alta_Envio]

    @IdentificadorEnvio nvarchar(20),
    @TipoEnvio nvarchar(20),
    @Fecha_Creacion datetime
AS
    insert into Envio (Identificador, Tipo, Fecha_Creacion, Enviado)
    values (@IdentificadorEnvio, @TipoEnvio, @Fecha_Creacion, 0)
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Login_Usuario]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Login_Usuario]

    @Usuario nvarchar(20)

AS
    select * from Usuarios where Nombre = @Usuario
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Identificador_Linea]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Identificador_Linea]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2)
AS
    select Identificador, fecha_Entrada from Linea where Numero_Documento = @Pedido and Sufijo = @Sufijo
    order by fecha_Entrada desc

    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_EnviosNoConsolidados]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_EnviosNoConsolidados]

    @IdentificadorEnvio nvarchar(20)
AS
    select * from Envio where enviado = 0 and Identificador like @IdentificadorEnvio + '%'
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Envios_Busq]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Envios_Busq]

    @IdentificadorEnvio nvarchar(20)
AS
    select * from Envio where Identificador like @IdentificadorEnvio + '%'

```

```

RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Envio]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Envio]

    @IdentificadorEnvio nvarchar(20)

AS

    select * from Envio where Identificador = @IdentificadorEnvio
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Todos]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Todos]

    @Pedido nvarchar(13)
AS

    select * from Pedido where Numero_Documento like @Pedido + '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Reparables_vuelta]  Script Date: 05/14/2012
18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Reparables_vuelta]

    @Pedido nvarchar(13)
AS

    SELECT      Linea.Identificador, Linea.Numero_Documento, Linea.Linea, Linea.Fecha_Entrada, Linea.Sufijo
FROM          Linea INNER JOIN
              Pedido ON Linea.Numero_Documento = Pedido.Numero_Documento
WHERE         (Linea.Identificador = 'XDC') AND (Pedido.Consolidado = 1) and Linea.Numero_Documento like @Pedido
              + '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Pendientes]  Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Pendientes]

    @Pedido nvarchar(13)
AS

    SELECT      Linea.Identificador, Linea.Numero_Documento, Linea.Linea, Linea.Fecha_Entrada, Linea.Sufijo
FROM          Linea LEFT OUTER JOIN
              Pedido ON Linea.Numero_Documento = Pedido.Numero_Documento AND Linea.Sufijo = Pedido.Sufijo
WHERE         (Pedido.Numero_Documento IS NULL) and Linea.Numero_Documento like @Pedido + '%'
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_NoProcesados]  Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_NoProcesados]

    @Pedido nvarchar(13)
AS
    select * from Pedido where Numero_Documento like @Pedido + '%' and Consolidado is null
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Usuarios]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Usuarios]

    @Usuario nvarchar(20)
AS
    select * from Usuarios where Nombre like @Usuario+ '%'
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Usuario_ByNombre]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Usuario_ByNombre]

    @Usuario nvarchar(20)
AS
    select * from Usuarios where Nombre = @Usuario
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimoEnvio]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_UltimoEnvio]
/*
(
    @parameter1 int = 5,
    @parameter2 datatype OUTPUT
)
*/
AS
        SELECT      TOP (1) Identificador, Tipo, Fecha_Creacion AS Expr1
        FROM          Envio
        ORDER BY Expr1 DESC

    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimoCargamento]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```

CREATE PROCEDURE [dbo].[sp_Obtener_UltimoCargamento]
/*
(
@parameter1 int = 5,
@parameter2 datatype OUTPUT
)
*/
AS
    SELECT          TOP (1) Identificador, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura, Tipo,
Fecha_Creacion AS Expr1
    FROM            Cargamento
    ORDER BY Expr1 DESC
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Detalle_Cargamento]    Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Detalle_Cargamento]

    @IdentificadorCargamento nvarchar(20)

AS
    select * from Cargamento where Identificador = @IdentificadorCargamento
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Contenido_Envio]    Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Contenido_Envio]

    @IdentificadorEnvio nvarchar(20)

AS
SELECT      Cargamento.*
FROM        Cargamento INNER JOIN
            Envio ON Cargamento.Envio = Envio.Identificador and Cargamento.Envio = @IdentificadorEnvio
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Cargamento]    Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Cargamento]

    @identificador nvarchar(20),
    @tipo nvarchar(20),
    @peso float,
    @volumen float,
    @altura float,
    @anchura float,
    @fecha_Creacion datetime,
    @Estado bit

AS

```

```

        Insert into Cargamento(identificador,Tipo,Fecha_Creacion,Estado,Peso,Volumen,Altura,Anchura)

        values(@identificador,@tipo,@fecha_Creacion,@Estado,@peso,@volumen,@altura,@anchura)
    RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Cargamento_En_Envio]    Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Cargamento_En_Envio]
    @IdentificadorEnvio nvarchar(20),
    @IdentificadorCargamento nvarchar(20)

AS
update Cargamento set Envio = @IdentificadorEnvio where Identificador = @IdentificadorCargamento
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cargamentos_NoInsertados]    Script Date: 05/14/2012
18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cargamentos_NoInsertados]

    @IdentificadorCargamento nvarchar(20)

AS

    SELECT      Identificador, Envio, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura, Tipo
FROM      Cargamento
WHERE      (Envio IS NULL) and Identificador like @IdentificadorCargamento + '%' and Cargamento.Estado = 0
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cargamentos]    Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cargamentos]

    @Identificador nvarchar(20),
    @Estado nvarchar(20)

AS
if @Estado = 'Abiertos'
begin
    select * from Cargamento where Identificador like @Identificador + '%' and Estado =1
end

if @Estado = 'Cerrados'
begin
    select * from Cargamento where Identificador like @Identificador + '%' and Estado = 0
end

if @Estado = 'Todos'
begin
    select * from Cargamento
end
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Incidencias]    Script Date: 05/14/2012 18:52:50 *****/

```



```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Incidencias]

    @Pedido nvarchar(13),
    @Sufijo nvarchar(2)
AS
    select * from Incidencias where Pedido = @Pedido and Sufijo = @Sufijo
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Cargamento_Pedido]    Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Sacar_Cargamento_Pedido]

    @IdentificadorEnvio nvarchar(20),
    @IdentificadorCargamento nvarchar(20)
AS
    Update Cargamento set Envio = null where Identificador =@IdentificadorCargamento
RETURN
GO
/***** Object: Table [dbo].[Caja]    Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Caja](
    [Identificador] [nvarchar](20) NOT NULL,
    [Cargamento] [nvarchar](20) NULL,
    [Fecha_Creacion] [datetime] NOT NULL,
    [Estado] [bit] NULL,
    [Peso] [float] NULL,
    [Volumen] [float] NULL,
    [Altura] [float] NULL,
    [Anchura] [float] NULL,
    [Tipo] [nvarchar](20) NULL,
    [Descripcion] [nvarchar](150) NULL,
    CONSTRAINT [PK_Caja] PRIMARY KEY CLUSTERED
(
    [Identificador] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Abrir_Cargamento]    Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Abrir_Cargamento]

    @Identificador nvarchar(20)
AS
    update Cargamento set Estado = 1 where Identificador = @Identificador
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Incidencia]    Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Incidencia]
@Pedido nvarchar(13),
@Sufijo nvarchar(2),
@tipoIncidencia nvarchar(20),
@observaciones nvarchar(150),
@usuario nvarchar(20),
@fecha_Creacion datetime
AS
    insert into Incidencias (Pedido,Sufijo,Tipo,Observaciones,Usuario,Fecha_Creacion)
        values (@Pedido,@Sufijo,@tipoIncidencia,@observaciones,@usuario,@fecha_Creacion)
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Cerrar_Cargamento]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cerrar_Cargamento]

    @Identificador nvarchar(20)
AS
    update Cargamento set Estado = 0 where Identificador = @Identificador
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Cerrar_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Cerrar_Caja]

    @Identificador nvarchar(20)
AS
    update Caja set Estado = 0 where Identificador = @Identificador
    RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Abrir_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Abrir_Caja]

    @Identificador nvarchar(20)
AS
    update Caja set Estado = 1 where Identificador = @Identificador
    RETURN
GO
/***** Object: Table [dbo].[Pedido_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Pedido_Caja](
    [Caja] [nvarchar](20) NOT NULL,
    [Pedido] [nvarchar](13) NOT NULL,
    [Sufijo] [nvarchar](2) NOT NULL,
    [Fecha_Inserccion] [datetime] NULL,
    [DivisionesTotales] [int] NULL,

```

```

        [NumeroDivision] [int] NOT NULL,
CONSTRAINT [PK_Pedido_Caja] PRIMARY KEY CLUSTERED
(
        [Caja] ASC,
        [Pedido] ASC,
        [Sufijo] ASC,
        [NumeroDivision] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_UltimaCaja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_UltimaCaja]
/*
(
        @parameter1 int = 5,
        @parameter2 datatype OUTPUT
)
*/
AS
        SELECT      TOP (1) Identificador, Cargamento, Fecha_Creacion, Estado, Peso, Volumen, Altura, Anchura,
        Tipo, Descripcion, Fecha_Creacion AS Expr1
        FROM          Caja
        ORDER BY Expr1 DESC

        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Sacar_Caja]

        @IdentificadorCargamento nvarchar(20),
        @IdentificadorCaja nvarchar(20)
AS
        update Caja set Cargamento =null where Identificador = @IdentificadorCaja
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cajas_NoInsertadas]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cajas_NoInsertadas]
        @IdentificadorCaja nvarchar(20)
AS
        SELECT      Caja.Identificador, Caja.Cargamento, Caja.Fecha_Creacion, Caja.Estado, Caja.Peso,
        Caja.Volumen, Caja.Altura, Caja.Anchura, Caja.Tipo, Caja.Descripcion
        FROM          Caja LEFT OUTER JOIN
        Cargamento ON Caja.Cargamento = Cargamento.Identificador
        WHERE          (Caja.Estado = 0) AND (Cargamento.Identificador IS NULL) and Caja.Identificador like
        @IdentificadorCaja + '%'
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Cajas]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Cajas]

    @Identificador nvarchar(20),
    @Estado nvarchar(20)

AS
if @Estado = 'Abiertas'
begin
    select * from Caja where Identificador like @Identificador + '%' and Estado = 1
end

if @Estado = 'Cerradas'
begin
    select * from Caja where Identificador like @Identificador + '%' and Estado = 0
end

if @Estado = 'Todas'
begin
    select * from Caja
end
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Caja]

    @IdentificadorCargamento nvarchar(20),
    @IdentificadorCaja nvarchar(20)

AS
    update Caja set Cargamento = @IdentificadorCargamento where Identificador = @IdentificadorCaja
RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Alta_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Alta_Caja]

    @identificador nvarchar(20),
    @tipo nvarchar(20),
    @peso float,
    @volumen float,
    @altura float,
    @anchura float,
    @descripcion nvarchar(150),
    @fecha_Creacion datetime,
    @Estado bit

AS

    Insert into Caja(identificador,Tipo,Fecha_Creacion,Estado,Peso,Volumen,Altura,Anchura,Descripcion)

    values(@identificador,@tipo,@fecha_Creacion,@Estado,@peso,@volumen,@altura,@anchura,@descrip
cion)

```

```

        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Contenido_Cargamento]  Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Contenido_Cargamento]
@Identificador nvarchar(20)

AS
        select * from Caja where Cargamento = @Identificador
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Detalle_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Detalle_Caja]

        @IdentificadorCaja nvarchar(20)

AS
        select * from Caja where Identificador = @IdentificadorCaja
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Sacar_Pedido_Caja]  Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Sacar_Pedido_Caja]

        @IdentificadorPedido nvarchar(13),
        @Sufijo nvarchar(2)

AS
        delete from Pedido_Caja where Pedido = @IdentificadorPedido and Sufijo = @Sufijo
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Ultimo_PedidoDiv]  Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Ultimo_PedidoDiv]

        @IdentificadorPedido nvarchar(13)

AS
        select * from Pedido_Caja where Pedido = @IdentificadorPedido order by NumeroDivision DESC
        RETURN
GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_NoInsertados]  Script Date: 05/14/2012 18:52:50
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_NoInsertados]

```

```

@Pedido nvarchar(13)

AS
SELECT      Pedido.Numero_Documento, Pedido.Sufijo, Pedido.Consolidado, Pedido.Fecha_Consolidacion,
Pedido.Fecha_Recepcion, Pedido.Fecha_Estimada

FROM        Pedido LEFT OUTER JOIN
            Pedido_Caja ON Pedido.Sufijo = Pedido_Caja.Sufijo AND Pedido_Caja.Pedido =
Pedido.Numero_Documento

WHERE       (Pedido_Caja.Pedido IS NULL) and Pedido.Numero_Documento like @Pedido + '%'
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Pedidos_Envio] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Pedidos_Envio]
    @IdentificadorEnvio nvarchar(20)
AS
    SELECT      Pedido.Numero_Documento, Pedido.Sufijo, Pedido.Consolidado, Pedido.Fecha_Consolidacion,
Pedido.Fecha_Recepcion, Pedido.Fecha_Estimada
FROM        Caja INNER JOIN
            Cargamento ON Caja.Cargamento = Cargamento.Identificador INNER JOIN
            Envio ON Cargamento.Envio = Envio.Identificador INNER JOIN
            Pedido_Caja ON Caja.Identificador = Pedido_Caja.Caja INNER JOIN
            Pedido ON Pedido_Caja.Pedido = Pedido.Numero_Documento AND Pedido_Caja.Sufijo = Pedido.Sufijo
WHERE       (Cargamento.Envio = @IdentificadorEnvio)
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Obtener_Datos_Caja] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Obtener_Datos_Caja]
    @IdentificadorCaja nvarchar(20)
AS
    SELECT      Caja.Identificador, Caja.Cargamento, Caja.Fecha_Creacion, Caja.Estado, Caja.Peso,
Caja.Volumen, Caja.Altura, Caja.Anchura, Caja.Tipo, Caja.Descripcion,
            Pedido_Caja.Caja, Pedido_Caja.Pedido, Pedido_Caja.Sufijo, Pedido_Caja.Fecha_Inserccion,
Pedido_Caja.DivisionesTotales, Pedido_Caja.NumeroDivision
FROM        Caja INNER JOIN
            Pedido_Caja ON Caja.Identificador = Pedido_Caja.Caja
WHERE       (Pedido_Caja.Caja = @IdentificadorCaja)
RETURN

GO
/***** Object: StoredProcedure [dbo].[sp_Insertar_Pedido_En_Caja] Script Date: 05/14/2012 18:52:50 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[sp_Insertar_Pedido_En_Caja]
    @IdentificadorCaja nvarchar(20),
    @IdentificadorPedido nvarchar(50),
    @IdentificadorSufijo nvarchar(2),
    @Fecha_Inserccion datetime,
    @NumeroTotalPartes int,
    @Parte int

```

AS

```
Insert Into Pedido_Caja (Caja, Pedido,Sufijo,Fecha_Inserccion,DivisionesTotales,NumeroDivision)

values(@IdentificadorCaja,@IdentificadorPedido,@IdentificadorSufijo,@Fecha_Inserccion
,@NumeroTotalPartes,@Parte)
RETURN
GO
/***** Object: ForeignKey [FK_Incidencias_Pedido]  Script Date: 05/14/2012 18:52:44 *****/
ALTER TABLE [dbo].[Incidencias] WITH CHECK ADD CONSTRAINT [FK_Incidencias_Pedido] FOREIGN KEY([Pedido],
[Sufijo])
REFERENCES [dbo].[Pedido] ([Numero_Documento], [Sufijo])
GO
ALTER TABLE [dbo].[Incidencias] CHECK CONSTRAINT [FK_Incidencias_Pedido]
GO
/***** Object: ForeignKey [FK_Cargamento_Envio]  Script Date: 05/14/2012 18:52:50 *****/
ALTER TABLE [dbo].[Cargamento] WITH CHECK ADD CONSTRAINT [FK_Cargamento_Envio] FOREIGN KEY([Envio])
REFERENCES [dbo].[Envio] ([Identificador])
GO
ALTER TABLE [dbo].[Cargamento] CHECK CONSTRAINT [FK_Cargamento_Envio]
GO
/***** Object: ForeignKey [FK_Caja_Cargamento]  Script Date: 05/14/2012 18:52:50 *****/
ALTER TABLE [dbo].[Caja] WITH CHECK ADD CONSTRAINT [FK_Caja_Cargamento] FOREIGN KEY([Cargamento])
REFERENCES [dbo].[Cargamento] ([Identificador])
GO
ALTER TABLE [dbo].[Caja] CHECK CONSTRAINT [FK_Caja_Cargamento]
GO
/***** Object: ForeignKey [FK_Pedido_Caja_Caja]  Script Date: 05/14/2012 18:52:50 *****/
ALTER TABLE [dbo].[Pedido_Caja] WITH CHECK ADD CONSTRAINT [FK_Pedido_Caja_Caja] FOREIGN KEY([Caja])
REFERENCES [dbo].[Caja] ([Identificador])
GO
ALTER TABLE [dbo].[Pedido_Caja] CHECK CONSTRAINT [FK_Pedido_Caja_Caja]
GO
/***** Object: ForeignKey [FK_Pedido_Caja_Pedido1]  Script Date: 05/14/2012 18:52:50 *****/
ALTER TABLE [dbo].[Pedido_Caja] WITH CHECK ADD CONSTRAINT [FK_Pedido_Caja_Pedido1] FOREIGN
KEY([Pedido], [Sufijo])
REFERENCES [dbo].[Pedido] ([Numero_Documento], [Sufijo])
GO
ALTER TABLE [dbo].[Pedido_Caja] CHECK CONSTRAINT [FK_Pedido_Caja_Pedido1]
GO
```